Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Average-Case Subset Balancing Problems

Xi Chen* Yaonan Jin† Tim Randolph‡ Rocco A. Servedio§

Abstract

Given a set of n input integers, the *Equal Subset Sum* problem asks us to find two distinct subsets with the same sum. In this paper we present an algorithm that runs in time $O^*(3^{0.387n})$ in the average case, significantly improving over the $O^*(3^{0.488n})$ running time of the best known worst-case algorithm [MNPW19] and the Meet-in-the-Middle benchmark of $O^*(3^{0.5n})$.

Our algorithm generalizes to a number of related problems, such as the "*Generalized Equal Subset Sum*" problem, which asks us to assign a coefficient c_i from a set C to each input number x_i such that $\sum_i c_i x_i = 0$. Our algorithm for the average-case version of this problem runs in time $|C|^{(0.5-c_0/|C|)n}$ for some positive constant c_0, whenever $C = \{0, \pm 1, \ldots, \pm d\}$ or $\{\pm 1, \ldots, \pm d\}$ for some positive integer d (with runtime $O^*(|C|^{0.45n})$ when $|C| < 10$). Our results extend to the problem of finding "nearly balanced" solutions in which the target is a not-too-large nonzero offset τ.

Our approach relies on new structural results that characterize the probability that $\sum_i c_i x_i = \tau$ has a solution $c \in C^n$ when x_i's are chosen randomly; these results may be of independent interest. Our algorithm is inspired by the "representation technique" introduced by Howgrave-Graham and Joux [HGJ10]. This requires several new ideas to overcome preprocessing hurdles that arise in the representation framework, as well as a novel application of dynamic programming in the solution recovery phase of the algorithm.

1 Introduction

The Subset Sum problem and its variants are among the most famous NP-complete problems, and the question of whether Subset Sum can be solved in time $O^*(2^{(0.5-\delta)n})$[1] for some constant $\delta > 0$ is one of the major questions in exact algorithms. Despite many attempts to find a "truly faster" exact algorithm, Horowitz and Sahni's classic $O^*(2^{0.5n})$-time Meet-in-the-Middle algorithm for worst-case inputs remains the benchmark after almost 50 years [HS74].

In recent years, the apparent difficulty of the Subset Sum problem has fueled work on variants and related settings. In 2010, [HGJ10] made a significant breakthrough by showing that Subset Sum could be solved in time $O^*(2^{0.337n})$ in the average case under a reasonable heuristic assumption.[2] Subsequent works, including [BCJ11, Böh11, BBSS20], refined what became known as the *representation technique*, and whittled the average-case exponent down to $O^*(2^{0.283n})$ under heuristic assumptions. This new technique inspired a flurry of results in related settings, including better time-space tradeoffs [AKKM13, DDKS12], a faster polynomial-space algorithm [BGNV18], and fast algorithms for large classes of sufficiently "random-like" instances [AKKN15, AKKN16].

In 2019, [MNPW19] used the representation technique to establish an $O^*(3^{0.488n})$-time *worst-case* algorithm for *Equal Subset Sum*, the Subset Sum variant that asks for two different input subsets with the same sum. This resolved an open question of Woeginger ([Woe08]), who observed that the Meet-in-the-Middle approach to Equal Subset Sum runs in time $O^*(3^{0.5n})$ and asked whether this was a barrier for Equal Subset Sum analogous to the $O^*(2^{0.5n})$ runtime barrier for Subset Sum. Equal Subset Sum is also closely related to the Number Balancing problem of [KK82], which can be thought of as the optimization version of Equal Subset Sum.

Both Subset Sum and Equal Subset Sum are special cases of the following more general problem: given a multiset of input integers, find a linear combination using only coefficients from a small set C that achieves a specified target value. Writing $[a : b]$ to denote the set of integers $\{a, a+1, \ldots, b\}$, we define:

*Columbia University. Email: xichen@cs.columbia.edu
†Columbia University. Email: yj2552@columbia.edu
‡Columbia University. Email: t.randolph@columbia.edu
§Columbia University. Email: rocco@cs.columbia.edu

[1] We use $O^*(\cdot)$ notation to suppress factors polylogarithmic in the argument of the function; so the notation "$O^*(2^{0.5n})$" suppresses $\mathrm{poly}(n)$ factors.

[2] Although the original paper claims a running time of $O^*(2^{0.311n})$, a correction of the original analysis gives this $O^*(2^{0.337n})$ runtime; see [BCJ11, Section 2.2] for details.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Problem 1: Generalized Subset Sum (GSS)

Input. An input range bound M, an input vector $\vec{x} = (x_1, \ldots, x_n) \in [0 : M-1]^n$, a set $C \subset \mathbb{Z}$ of allowed coefficients, and a target integer τ.
Output. A coefficient vector $\vec{c} \in C^n$ such that $\vec{c} \cdot \vec{x} = \tau$, if one exists.

It is natural to compare the runtime of GSS algorithms to the size of the search space $|C|^n$. There is a straightforward generalization of Horowitz and Sahni's Meet-in-the-Middle Algorithm for GSS: partition the input into two vectors $\vec{x}_1$ and $\vec{x}_2$ of length $n/2$ each, list all partial linear combinations $\vec{c}_1 \cdot \vec{x}_1$ and $\vec{c}_2 \cdot \vec{x}_2$ by enumerating all $\vec{c}_1, \vec{c}_2 \in C^{n/2}$, and search the lists for a pair of linear combinations such that $\vec{c}_1 \cdot \vec{x}_1 + \vec{c}_2 \cdot \vec{x}_2$ achieves the target τ. This can be done in time $O^*(|C|^{n/2})$ by sorting the lists and using two pointers that walk from opposite ends of the two lists, seeking a pair that sums to the target. The $C = \{0, 1\}$ case of this algorithm is Horowitz and Sahni's Meet-in-the-Middle algorithm for Subset Sum, and the $C = \{0, \pm 1\}$ case (with target $\tau = 0$ and the solution $\vec{c} = \vec{0}$ disallowed) is the Meet-in-the-Middle algorithm for Equal Subset Sum.

In light of the broad body of work that has been done on the average-case version of the original Subset Sum problem, it is natural to consider the *average-case* GSS problem in which the input vector $\vec{x}$ is uniformly random over $[0 : M-1]^n$; this average-case GSS problem is the subject of the current paper. There are two natural average-case variants of the GSS problem: in the first variant, the target value is obtained by sampling a "hidden" solution, and hence every instance of this average-case problem variant admits a solution. This variant of the problem is motivated by cryptographic applications, and corresponds to the average-case variant of Subset Sum that was studied by [HGJ10, BCJ11] and others; we refer to it as the "cryptographic version" of the average-case GSS problem. In the second version, which is the one we consider in this work, the target is a fixed value which does not depend on the draw of random input $\vec{x}$. This version, which we refer to as the "balancing version" (see the formal definition below), has more of the flavor of Equal Subset Sum. Since both yes-instances and no-instances are possible for this variant, a natural goal that arises in its study is to understand the probability (as a function of the various parameter settings) that a solution exists. Structural questions of this type have in fact been the subject of considerable study for the special case of $C = \{\pm 1\}$; see for example [BCP01, Lue98].

1.1 Our Results We consider two families of symmetric coefficient sets, with and without zero. For a fixed positive integer $d \in \mathbb{N}$, we define $C(d) = \{\pm 1, \pm 2, \ldots, \pm d\}$ and $C_0(d) = \{0, \pm 1, \pm 2, \ldots, \pm d\}$. These two sets cover a wide range of cases, including Equal Subset Sum. We note that while the natural coefficient set corresponding to Subset Sum is $C = \{0, 1\}$, an instance $\vec{x}$ with target τ can be easily converted to GSS on $C(1) = \{\pm 1\}$ by setting a new target $\tau' := 2\tau - \sum_{i \in [n]} x_i$.[3]

We consider algorithms for the balancing variant of average-case GSS over either $C = C(d)$ or $C_0(d)$. Given as input M, τ and $\vec{x} \in [0 : M-1]^n$, such an algorithm either returns "no solution" or a solution $\vec{c}$ that satisfies $\vec{c} \cdot \vec{x} = \tau$ (with $\vec{c} \neq \vec{0}$ when $C = C_0(d)$ and $\tau = 0$). We say an algorithm fails on $(M, \tau, \vec{x})$ if it returns "no solution" but indeed there is a solution; otherwise we say it succeeds.

Our main algorithmic result for (the balancing variant of) average-case GSS is as follows.

Theorem 1 (Algorithm for Average-Case GSS)**.** *Fix any $d \in \mathbb{N}_{\geq 1}$ and let $C = C(d)$ or $C_0(d)$. For any constant $\zeta > 0$, there is a randomized algorithm for average-case GSS with running time*

$$O^*(|C|^{\Lambda(|C|)n + \zeta n})^4 \qquad \text{where} \quad \Lambda(z) = \max \begin{cases} 1 - \frac{z+1}{2z} \log_z(z+1) + \frac{1}{z} \log_z(2) \\ \frac{2}{3} - \frac{z+1}{3z} \log_z\left(\frac{z+1}{2}\right) \end{cases}.$$

Given any M and τ with $|\tau| = o(nM)$, the algorithm succeeds on $(M, \tau, \vec{x})$ with probability at least $1 - e^{-\Omega(n)}$ when $C = C_0(d)$ and with probability at least $1 - o_n(1)$ when $C = C(d)$, over the draw of $\vec{x} \sim [0 : M-1]^n$ and the randomness of the algorithm.

[3] In fact, our results extend to all scale multiples and translations of $C(d)$ and $C_0(d)$. For details, refer to Section 5.1.

[4] Note that the runtime of our algorithm is independent of the input bound M. This occurs because the probability of a 'yes' instance is exponentially small when $M = 2^{\omega(n)}$ (refer to Theorems 3 and 4).

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

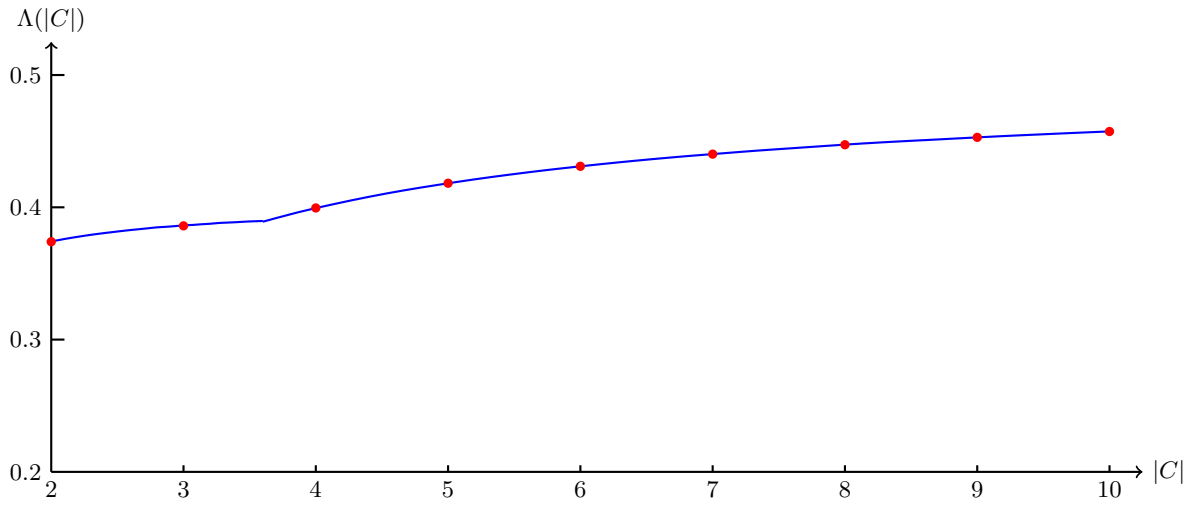


Figure 1: Plot of Λ. The red points plot $\Lambda(z)$ for $z \in [2:10]$.

$d \in \mathbb{N}_{\geq 1}$		1	2	3	5	10
Runtime	$C(d)$	$O^*(\lvert C\rvert^{0.375n})^\dagger$	$O^*(\lvert C\rvert^{0.400n})$	$O^*(\lvert C\rvert^{0.432n})$	$O^*(\lvert C\rvert^{0.458n})$	$O^*(\lvert C\rvert^{0.479n})$
	$C_0(d)$	$O^*(\lvert C\rvert^{0.387n})^\ddagger$	$O^*(\lvert C\rvert^{0.419n})$	$O^*(\lvert C\rvert^{0.441n})$	$O^*(\lvert C\rvert^{0.462n})$	$O^*(\lvert C\rvert^{0.480n})$

†Our $C(1)$ case differs from the "cryptographic" average-case Subset Sum problem considered in [HGJ10, BCJ11, Böh11, BBSS20], because we consider the "balancing" problem with a fixed offset (for which a solution may or may not exist); see the Introduction for further discussion.

‡This runtime for $C_0(1)$ should be contrasted with the $O^*(|C|^{0.488n})$-time worst-case runtime due to [MNPW19].

Table 1: Runtime of our algorithm for average-case GSS on various coefficient sets.

We note that $\Lambda(z) = 0.5 - \Omega(1/z)$, and thus our algorithm beats the Meet-in-the-Middle runtime of $O^*(|C|^{0.5n})$ by an exponential margin for every constant $|C|$. Figure 1 plots the function Λ and Section 1.1 lists our algorithm's runtime on various coefficient sets.

As a special case of Theorem 1 we obtain an average-case algorithm for Equal Subset Sum that significantly improves on the worst-case $O^*(3^{0.488n})$ runtime of [MNPW19]:

Corollary 1 (Algorithm for Average-Case Equal Subset Sum)**.** *There exists an algorithm that solves Average-Case Equal Subset Sum in time* $O^*(3^{0.387n})$ *with success probability* $1 - e^{-\Omega(n)}$.

Our algorithm has the additional property that it runs faster on *dense* instances, i.e., ones for which M is substantially less than $|C|^n$. Intuitively, this is possible because in this regime there are likely to be many solutions.

Theorem 2 (Average-Case GSS on Dense Instances)**.** *Fix* $d \in \mathbb{N}_{\geq 1}$, $C = C(d)$ *or* $C = C_0(d)$, $M = |C|^{\alpha n + o(n)}$ *for some* $\alpha \in (0,1)$ *and an offset* τ *with* $|\tau| = o(Mn)$. *For any constant* $\zeta > 0$, *there exists an algorithm that solves average-case GSS in time*

$$O^*(|C|^{\alpha\Lambda(|C|)n + \zeta n}),$$

where Λ *is defined as in Theorem 1. The algorithm succeeds with probability at least* $1 - e^{-\Omega(n)}$ *for* $C = C_0(d)$ *and with probability at least* $1 - o_n(1)$ *for* $C = C(d)$.

Crucial ingredients underlying our algorithms are new structural results on the probability that random GSS instances have solutions. In the context of previous work on closely related questions [BCP01, Lue98], we believe that our new structural results may be of independent interest; we explain our new structural results and contrast them with prior work below.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

1.2 Our Techniques To explain our structural and algorithmic results, we begin with some intuition for the distribution of sums that are achievable using coefficient set C. Consider a uniform random draw of the input vector $\vec{\boldsymbol{x}}$ from $[0 : M-1]^n$. Since the coefficient set C is symmetric about 0, all elements of the random set $\boldsymbol{S}_n := \{\vec{c} \cdot \vec{\boldsymbol{x}} : \vec{c} \in C^n\}$ have magnitude $O(Mn)$, and hence intuitively $\boldsymbol{S}_n$ is "tightly concentrated around the origin". Consequently, when the offset τ is not too large (the case of interest for us) it is natural to expect that a solution $\vec{c} \cdot \vec{\boldsymbol{x}} = \tau$ is likely to exist if $M \ll |C|^n$ and is unlikely to exist if $M \gg |C|^n$. Indeed, a simple counting argument establishes this for the case that $M \gg |C|^n$, but substantially more work is required to rigorously confirm the above intuition for the $M \ll |C|^n$ case. This is accomplished by our structural results, which we describe next.

1.2.1 Structural Results Our structural results for coefficient set $C_0(d)$ (including zero) and for $C(d)$ (not including zero) are established using very different techniques. Consider the case of $C = C_0(d)$ first. We are able to show that for any positive constant $\epsilon > 0$, given $M \leq |C|^{(1-\epsilon)n}$ and any fixed integer offset τ with $|\tau| = o(Mn)$, the probability (over a uniform random $\vec{\boldsymbol{x}} \sim [0 : M-1]^n$) that there exists a solution $\vec{c} \cdot \vec{\boldsymbol{x}} = \tau$ with $\vec{c} \in C^n$ is *exponentially* close to 1. See Theorem 4 for a detailed statement of this result. This extremely high probability that there exists a solution translates directly into the extremely high success probability of our algorithms in Theorem 1 and Theorem 2.

To establish this structural result, we employ a novel proof strategy that aligns with the intuition sketched earlier and is based on an iterative analysis. We define random sets $\boldsymbol{S}_1, \boldsymbol{S}_2, \ldots$ where the set $\boldsymbol{S}_\ell$ for $\ell \in [n]$ is given by $\boldsymbol{S}_\ell := \{\vec{c}' \cdot \vec{\boldsymbol{x}}' : \vec{c}' \in C^\ell\}$ with a uniform random $\vec{\boldsymbol{x}}' \sim [0 : M-1]^\ell$. We then analyze how these set sizes $|\boldsymbol{S}_\ell|$ increase as a function of ℓ by thinking of $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots$ as being drawn in succession. We first argue that with very high probability $|\boldsymbol{S}_\ell|$ increases rapidly with ℓ until, at some value $\boldsymbol{L}_1 < n$, it reaches a point at which it is dense on at least one "large" interval that is "close to" the origin. We then argue that at this point it suffices to draw a few more elements to ensure that some *partial solution* $\sum_{i \in [\boldsymbol{L}_2]} c_i \boldsymbol{x}_i$ will hit the offset τ with very high probability, where $\boldsymbol{L}_2 \geq \boldsymbol{L}_1$ and $\boldsymbol{L}_2$ is still less than n. The remaining elements $\boldsymbol{x}_{\boldsymbol{L}_2+1}, \ldots, \boldsymbol{x}_n$ are simply assigned the 0 coefficient to complete the overall solution. The detailed proof is given in Section 3.4.

Turning to the case of coefficient set $C = C(d)$, it is clear that the absence of the 0 coefficient is a fundamental obstacle to the previous approach, and indeed we do not know how to achieve an exponentially high success probability for $C = C(d)$. Instead, to handle this case we use a very different proof strategy that extends the approach of [BCP01], who analyzed the $C(d)$ case for $d = 1$. Their analysis (see [BCP01, Theorem 2.1]) shows that for any $M \leq 2^{(1-\epsilon)n}$, the probability that a uniform random input $\vec{\boldsymbol{x}} \sim [0 : M-1]^n$ admits a "perfect" solution $\vec{c} \in \{\pm 1\}^n$ is $1 - o_n(1)$, where a "perfect" solution is one satisfying $\vec{c} \cdot \vec{\boldsymbol{x}} = 1$ if $\sum_{i \in [n]} x_i$ is odd and satisfying $\vec{c} \cdot \vec{\boldsymbol{x}} = 0$ if $\sum_{i \in [n]} x_i$ is even.[5]

We extend the [BCP01] analysis and establish a similar result for general coefficient sets $C = C(d)$ for arbitrary $d > 1$. We show that for any constant $d > 1$, if $M \leq |C|^{(1-\epsilon)n}$ then given any fixed integer offset τ with $|\tau| = o(Mn)$, the probability (over a uniform draw of $\vec{\boldsymbol{x}}$ from $[0 : M-1]^n$) that there exists a solution $\vec{c} \cdot \vec{\boldsymbol{x}} = \tau$ with $\vec{c} \in C^n$ is $1 - o_n(1)$. At a high level, our analysis establishing this follows the arguments of [BCP01]; we write the number of solutions as a random integral over all coefficient vectors, then bound suitable integrals to characterize the first moment and upper bound the second moment of the relevant random variable. The analysis requires considerable attention to detail (the domain of integration needs to be broken into three different regions, with different arguments required for each region); see Theorem 3 for a detailed statement and Appendix C for the proof.

Remark 1 (Comparison with [Lue98]). It is interesting to compare our structural results Theorem 3 and Theorem 4 with the main structural result of [Lue98]. That work analyzes the probability that given a real-valued vector $\vec{\boldsymbol{X}} = (\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n)$ drawn uniformly at random from $[-1, 1]^n$, there exists a coefficient vector $\vec{\delta} \in \{0, 1\}^n$ such that $\vec{\delta} \cdot \vec{\boldsymbol{X}}$ lies within a small range $\pm\eta$ of a specified target value $z \in [-\frac{1}{2}, \frac{1}{2}]$. By scaling the $\boldsymbol{X}_i$'s, [Lue98] implies that if $M \leq \kappa^n$ where $\kappa = 2^{1/(2+2\ln 2)} \approx 1.227$, then for any target $\tau \in [-\frac{M}{2}, \frac{M}{2}]$, with exponentially high probability over a uniform draw of $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ each from the continuous interval $[-M, M]$, there exists a solution $\vec{c} \in C(1)^n = \{\pm 1\}^n$ for which $|\vec{c} \cdot \vec{x} - \tau| \leq 1/2$. While this result has a similar flavor to our Theorem 3, our Theorem 4, and [BCP01, Theorem 2.1], it is incomparable to all of them (even apart from the fact that it deals

[5] [BCP01] further established a range of more refined structural results, but this is the most relevant result for our purposes.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

with continuous rather than discrete random variables). The exponentially high probability bound that it gives is similar to the Theorem 4 bound and is better than the $1 - o_n(1)$ probability bound of [BCP01, Theorem 2.1], but the quantitative bound $M \leq \kappa^n$ that it requires is worse than the bound allowed by [BCP01, Theorem 2.1]. We note that unlike the result implicit in [Lue98], both Theorem 3 and Theorem 4 apply to general coefficient sets C, and that both Theorem 3, Theorem 4 and [BCP01, Theorem 2.1] allow the "correct" range of values for M, i.e., M is allowed to be as large as $|C|^{(1-\epsilon)n}$. (Having this "correct" range of values for M is essential for our ultimate purpose of developing average-case GSS algorithms. Otherwise, there would be a wide range of values for M to which no strong structural result would apply.)

1.2.2 Algorithmic Results Intuitively, our structural results tell us the parameter settings for which average-case GSS instances are likely to have solutions and what those solutions will look like. This allows us to prove the correctness of our algorithm, which combines new preprocessing ideas with an approach based on the representation technique.

The Representation Technique. The representation technique works by constructing many partial *candidate solutions* and filtering them in a way that reduces the search space while retaining at least one way of constructing a solution. For example, the key observation of [HGJ10] is that any Subset Sum solution of size $n/2$ can be decomposed into $\Omega^*(2^{0.5n})$ "solution pairs", each consisting of two disjoint sets of size $n/4$. Since there are $\binom{n}{n/4} = \Theta^*(2^{0.811n})$ input subsets of size $n/4$, this means that (under sufficiently strong assumptions on the input) hashing these subsets in a way that ensures solution pairs hash to the same value reduces the search space to $O^*(2^{0.811n-0.5n}) = O^*(2^{0.311n})$ elements, from which a solution can be recovered using Meet-in-the-Middle.[6]

For C larger than $\{0, 1\}$, the representation approach considers partial candidate solutions that assign nonzero coefficients to some input elements. For example, [MNPW19] consider $C = \{0, \pm 1\}$ and partial candidate solutions that pick out approximately $n/6$ coefficients matched with each of 1 and -1. However, this approach works only for yes-instances with a significant number of 0 coefficients. We show how to circumvent the problem for general C by "translating" GSS instances so that any $c \in C$ can play the role of the 0 coefficient.

At a high level, since instances in which $M > |C|^{(1+\epsilon)n}$ are extremely unlikely to have a solution, we can simply ignore such instances. On the other extreme, we show how to "shrink" instances for which $M < |C|^{(1-\epsilon)n}$ for any constant ϵ while preserving the existence of a solution. (The "shrinking" also gives an improved runtime for our algorithm). Thus, we can focus on a narrow range of values for M. For such instances, we simulate a hash based on the *signature*, or partial sum, of a partial candidate solution in such a way that the two halves of a genuine solution are guaranteed to hash to the same value. We prove a result (Lemma 4) which guarantees that with high probability many partial solutions have distinct signatures and thus removes the need for any heuristic assumptions. Finally, we use a dynamic programming approach to simulate the signature-based hash, sample elements, and recover a solution. Section 4 presents our algorithm with preprocessing, correctness and runtime proofs.

1.3 Organization Section 2 establishes useful notation and preliminaries. Our structural results are in Section 3 and our algorithmic results are in Section 4. Finally, Section 5 gives average-case reductions from GSS to (Generalized) Number Balancing as well as versions of our structural results in this setting.

2 Notation and Preliminaries

Given an n-dimensional vector $\vec{x}$ and $T \subseteq [n]$, we write $\vec{x}_T$ to denote the $|T|$-dimensional vector obtained as the projection of $\vec{x}$ onto indices $i \in T$.

When written without a specified base, $\log(\cdot)$ denotes the base-2 logarithm.

Parameter assumptions. In the proofs below, we consider $M = |C|^{\Omega(n)}$. When $M = |C|^{o(n)}$, standard dynamic programming techniques solve GSS in subexponential time.

Big-O notation. An asterisk added to big-O notation (O^*, Ω^*, and Ω^*) indicates the suppression of factors

[6] Several challenges arise in making these ideas precise, including efficiently simulating the hash function and proving that randomness in the input results in candidate solutions that are favorably distributed with high probability. For a more detailed introduction to the representation technique for Subset Sum, see [BCJ11, Section 2.2].

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

polylogarithmic in the argument of the function. For example, $O^*(n)$ ignores polylog(n) factors and $O^*(2^n)$ ignores poly(n) factors.

Set notation. We write $[a:b]$ for the set of integers $\{a, a+1, \ldots, b\}$. This notation is simplified to $[a]$ for $\{1, 2, \ldots, a\}$. We write $\Sigma(S)$ as shorthand for the sum $\sum_{s\in S} s$, αS as shorthand for the multiset $\{\alpha s : s \in S\}$, and $S + \alpha$ as shorthand for the multiset $\{s + \alpha : s \in S\}$. For $\epsilon \in (0, 1)$ and $b \geq 1$, we write $a \in b^{1\pm\epsilon}$ to indicate $a \in [b^{1-\epsilon}, b^{1+\epsilon}]$.

Probability notation. Random variables like $\boldsymbol{x}_i$ are written in boldface. For a finite set of vectors S, we write "$\vec{\boldsymbol{x}} \sim S$" to indicate that vector $\vec{\boldsymbol{x}}$ is sampled uniformly at random from S.

Combinatorics. Recall the definition of multinomial coefficients:

$$\binom{n}{\alpha_1 n, \alpha_2 n, \ldots, \alpha_m n} := \frac{n!}{(\alpha_1 n)!(\alpha_2 n)! \ldots (\alpha_m n)!}$$

for nonnegative $\alpha_1, \alpha_2, \ldots, \alpha_m$ such that $\sum_{i\in[m]} \alpha_i = 1$ and $\alpha_i n$'s are integers for all i. Stirling's approximation tells us that $n! = \Theta^*(n^n e^{-n})$. Substituting yields the helpful approximation

$$\binom{n}{\alpha_1 n, \alpha_2 n, \ldots, \alpha_m n} = \Theta^*\Big(\prod_{i\in[m]} \alpha_i^{-\alpha_i n}\Big) = \Theta^*\big(2^{H(\alpha_1,\alpha_2,\ldots,\alpha_m)n}\big), \tag{2.1}$$

where $H(\alpha_1, \alpha_2, \ldots, \alpha_m) := -\sum_{i\in[m]} \alpha_i \log_2(\alpha_i)$ denotes the entropy function. We write $H(\alpha)$ for the binary entropy function $H(\alpha, 1-\alpha)$.

3 Structural Results

This section presents our structural results, which characterize when average-case GSS instances are likely to have a solution in terms of M, n, τ and $|C|$.

3.1 The $C = C(1)$ Case

The $C = C(1) = \{\pm 1\}$ case has been studied in previous work by Borgs, Chayes and Pittel [BCP01]. Their results precisely determine the parameters of a phase transition within a subexponential window around $2^n/\sqrt{n}$ and prove that solutions exist with probability $1 - o_n(1)$ for M below the window and $o_n(1)$ above. For our purposes, precisely pinning down the phase transition window is less important than establishing regions within which solutions are either very likely or very unlikely to exist, so we present a corollary of their analysis ([BCP01, Theorem 2.1]) which features a larger window but more flexibility in the offset. The proof can be found in Appendix A.

Corollary 2 (GSS Solution Probability on $C(1)$ [BCP01])**.** *Let $C = C(1)$ and fix any positive constant $\epsilon > 0$. For $\vec{\boldsymbol{x}} = (\boldsymbol{x}_i)_{i\in[n]} \sim [0 : M-1]^n$ and any integer τ satisfying $|\tau| = o(Mn)$, we have*

$$\Pr_{\vec{\boldsymbol{x}}}\Big[\exists \vec{c} \in C^n : \vec{\boldsymbol{x}} \cdot \vec{c} \in \{\tau, \tau+1\}\Big] \begin{cases} \geq 1 - o_n(1) & \text{if } M \leq |C|^{(1-\epsilon)n} \\ \leq 2|C|^n/M & \text{if } M \geq |C|^n. \end{cases}$$

Note that $\vec{\boldsymbol{x}} \cdot \vec{c}$ has the same parity as $\sum_{i\in[n]} \boldsymbol{x}_i$ for every $\vec{c} \in \{\pm 1\}^n$. So the parity of $\sum_{i\in[n]} \boldsymbol{x}_i$ determines a single possible target in $\{\tau, \tau+1\}$.

3.2 The $C = C(d)$, $d > 1$ Case

The proof of [BCP01] can be further extended to the case of $C(d)$ with a general $d > 1$. We prove the following theorem in Appendix C:

Theorem 3 (GSS Solution Probability for $C(d)$, $d > 1$)**.** *Let $C = C(d)$ for a fixed integer $d > 1$ and fix any constant $\epsilon > 0$. For $\vec{\boldsymbol{x}} \sim [0 : M-1]^n$ and any integer τ satisfying $|\tau| = o(Mn)$, we have*

$$\Pr_{\vec{\boldsymbol{x}}}\Big[\exists \vec{c} \in C^n : \vec{\boldsymbol{x}} \cdot \vec{c} = \tau\Big] \begin{cases} \geq 1 - o_n(1) & \text{if } M \leq |C|^{(1-\epsilon)n} \\ \leq |C|^n/M & \text{if } M \geq |C|^n. \end{cases}$$

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Generalizing the proof strategy of [BCP01] to $C(d)$ significantly complicates the analysis but does not substantially change the underlying intuition. Notably, [BCP01] confronted a parity issue in their case of $C(1)$: no solution summing to a target τ that has parity different from the sum of inputs is possible. Our analysis explains the parity issue as a result of constructive interference in the integrand of the solution-counting function and demonstrates that no such issues occur for $C(d)$ when $d > 1$ (compare the $\vec{\boldsymbol{x}} \cdot \vec{c} \in \{\tau, \tau+1\}$ in Corollary 2 and $\vec{\boldsymbol{x}} \cdot \vec{c} = \tau$ in Theorem 3).

This proof approach has the advantage that it neatly bounds the first and second moments of the number of solutions, not just the probability that a solution exists. However, the probability that a solution exists in the $M \le |C|^{(1-\epsilon)n}$ regime will correspond to the failure probability of our algorithm later, so it is desirable to have a sharper bound than the $1 - o_n(1)$ given by Corollary 2 and Theorem 3. We achieve this in the $C = C_0(d)$ case by using a new approach described in the next subsection.

3.3 The $C = C_0(d)$ Case When the coefficient set contains 0, we analyze the probability that a solution exists by considering a process in which a set of "achievable targets" is grown as input numbers $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots$ are revealed one by one. Once the set of achievable targets contains τ, the existence of a solution is ensured since any remaining input elements can be assigned the 0 coefficient. The structural result that we prove in this way is analogous to Theorem 3, but with exponentially small failure probability for small M. Moreover, the argument is shorter, more elementary, and more intuitive.

Theorem 4 (GSS Solution Probability for $C = C_0(d)$)**.** *Let $C = C_0(d)$ for some fixed $d \in \mathbb{N}$, and fix any constant $\epsilon > 0$. For $\vec{\boldsymbol{x}} \sim [0 : M-1]^n$ and any integer τ satisfying $|\tau| = o(Mn)$, we have*

$$\mathop{\mathbf{Pr}}_{\vec{\boldsymbol{x}}}\left[\exists\, \vec{c} \in C^n : \vec{\boldsymbol{x}} \cdot \vec{c} = \tau\right] \begin{cases} \ge 1 - e^{-\Omega(n)} & \text{if } M \le |C|^{(1-\epsilon)n} \\ \le |C|^n / M & \text{if } M \ge |C|^n. \end{cases}$$

3.4 Proof of Theorem 4 We first give the simple upper bound in the case when $M \ge |C|^n$, which is just a union bound. Any fixed *not-all-zero* $\vec{c} \in C^n$ has $c_i \ne 0$ for some $i \in [n]$. Conditioning on any outcome of $\vec{\boldsymbol{x}}_{[n]\setminus\{i\}}$ from $[0 : M-1]^{n-1}$, $\vec{\boldsymbol{x}} \cdot \vec{c} = \tau$ holds for at most one outcome of $\boldsymbol{x}_i$ and thus occurs with probability at most $1/M$. Union-bounding over all $\vec{c} \in C^n$ gives the claimed upper bound of $|C|^n/M$.

In the rest of the section, we focus on the case when $M \le |C|^{(1-\epsilon)n}$ for some positive constant $\epsilon > 0$. Let $d \in \mathbb{N}$ be a constant with $C = C_0(d)$, and let τ be the target integer with $|\tau| = o(Mn)$. For convenience we assume in the proof that $\tau \ne 0$; it will become clear at the end that the same proof works for $\tau = 0$. We start with some notation and a high-level overview of the proof.

Given $\vec{x} \in [0 : M-1]^\ell$, we define the set $\mathcal{S}(\vec{x})$ of *targets achievable with* $\vec{x}$ to be

$$\mathcal{S}(\vec{x}) := \left\{\vec{x} \cdot \vec{c} : \vec{c} \in C^\ell\right\}.$$

When $\ell = 0$ and $\vec{x}$ is the empty vector, $\mathcal{S}(\vec{x}) = \{0\}$ by default. Given $\vec{x} \in [0 : M-1]^n$, note that

$$\mathcal{S}(\vec{x}_{[1]}) \subseteq \mathcal{S}(\vec{x}_{[2]}) \subseteq \cdots \subseteq \mathcal{S}(\vec{x}_{[n]})$$

and $\vec{x}$ admits a solution if and only if $\mathcal{S}(\vec{x}_{[n]}) = \mathcal{S}(\vec{x})$ contains τ.[7] Let

$$m := \left\lceil \tau n + \log_{|C|} M \right\rceil \quad \text{and} \quad m' := \left\lceil \left(1 - \frac{\epsilon}{3}\right) n \right\rceil, \quad \text{with } \tau = \frac{\epsilon^2}{256 d^2 \ln |C|}.$$

Using $\log_{|C|} M \le (1-\epsilon)n$, we have $m' - m > \epsilon n/3$.

Our proof considers the evolution of the set $\boldsymbol{S}_\ell := \mathcal{S}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_\ell)$ as we draw each input element $\boldsymbol{x}_\ell \sim [0 : M-1]$ for $\ell = 1, \ldots, n$ sequentially. It proceeds in three steps:

1. Show that with high probability over the draw of the first m elements, $\boldsymbol{S}_m$ occupies a constant fraction of some length-M interval not too far from the origin. (Lemma 1).

[7] This is the reason why we assumed $\tau \ne 0$; when $\tau = 0$ we need the extra condition that $0 \in \mathcal{S}(\vec{x})$ can be obtained by a not-all-zero $\vec{c}$.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

2. Assuming the event in item 1 happens, show that with high probability over the draw of the next $m' - m$ elements, $\boldsymbol{S}_{m'}$ occupies a constant fraction of a length-M interval that contains τ. (Lemma 2).

3. Assuming the event in item 2 happens, show that $\tau \in \boldsymbol{S}_n$ with high probability over the draw of the last $n - m'$ elements.

Among these three steps, the first step is the most challenging.

Before stating Lemma 1, we give one more definition about what we meant by "not too far from the origin." Given $\vec{x} \in [0 : M-1]^\ell$, we define the set of *large* targets achievable with $\vec{x}$ as

$$\mathcal{B}(\vec{x}) := \Big\{ w \in \mathcal{S}(\vec{x}) : |w| \geq \epsilon n M/8 \Big\}.$$

The following simple claim shows that $\mathcal{B}(\vec{x})$ cannot be too large:

Claim 1. *For any $\ell \in [0:n]$ and $\vec{x} \in [0:M-1]^\ell$, we have*

$$|\mathcal{B}(\vec{x})| \leq |C|^\ell \cdot 2\exp\left(-\frac{\epsilon^2 n}{128 d^2}\right) = |C|^\ell \cdot 2|C|^{-2\tau n}.$$

Proof. The case with $\ell = 0$ is trivial given $\mathcal{B}(\vec{x}) = \emptyset$. For $\ell \geq 1$, let independent random variables $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_\ell$ be such that $\mathbf{Pr}[\boldsymbol{y}_i = c x_i] = 1/|C|$ for each $c \in C$. Letting $\boldsymbol{y} = \sum_{i \in [\ell]} \boldsymbol{y}_i$, we have

$$\frac{|\mathcal{B}(\vec{x})|}{|C|^\ell} \leq \mathop{\mathbf{Pr}}_{\boldsymbol{y}}\Big[|\boldsymbol{y}| \geq \epsilon n M/8\Big]$$

and the claim follows from an application of the Hoeffding inequality (and $\ell \leq n$). □

We prove Lemma 1 for the first step:

Lemma 1 (Dense Interval Lemma)**.** *With probability $1 - e^{-\Omega(n)}$ over $\vec{\boldsymbol{x}} \sim [0:M-1]^m$, $\mathcal{S}(\vec{\boldsymbol{x}}) \setminus \mathcal{B}(\vec{\boldsymbol{x}})$ occupies at least γ-fraction of some length-M interval with*

$$\gamma := \frac{\tau^2}{48|C|^2}, \tag{3.2}$$

i.e., there exists a length-M interval I such that $|I \cap (\mathcal{S}(\vec{\boldsymbol{x}}) \setminus \mathcal{B}(\vec{\boldsymbol{x}}))| \geq \gamma M$.

Proof. We use the following claim to establish the lemma:

Claim 2. *Let $\ell \in [m]$ and let $\vec{y} \in [0:M-1]^{\ell-1}$ be a vector such that $|I \cap (\mathcal{S}(\vec{y}) \setminus \mathcal{B}(\vec{y}))| < \gamma M$ for any length-M interval I. When $\boldsymbol{z} \sim [0:M-1]$, we have*

$$\frac{|\mathcal{S}(\vec{y} \circ \boldsymbol{z})|}{|\mathcal{S}(\vec{y})|} \geq (1 - \tau/4)|C|$$

with probability at least $1 - \tau/4$.

We delay the proof of the claim and use it to prove the lemma first. Consider the experiment of drawing $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m \sim [0:M-1]$ in turn, with $\boldsymbol{S}_\ell = \mathcal{S}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_\ell)$ and $\boldsymbol{B}_\ell = \mathcal{B}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_\ell)$ for each $\ell \in [0:m]$ (so $\boldsymbol{S}_0 = \{0\}$ and $\boldsymbol{B}_0 = \emptyset$). For each $\ell \in [m]$, let $\boldsymbol{\mathcal{X}}_\ell$ denote the indicator random variable that is set to 1 if either $|I \cap (\boldsymbol{S}_{\ell-1} \setminus \boldsymbol{B}_{\ell-1})| \geq \gamma M$ for some length-M interval I or

$$\frac{|\boldsymbol{S}_\ell|}{|\boldsymbol{S}_{\ell-1}|} \geq (1 - \tau/4)|C|. \tag{3.3}$$

Then conditioning on any outcome of $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{\ell-1})$ it follows from Claim 2 that the probability of $\boldsymbol{\mathcal{X}}_\ell = 1$ is at least $1 - \tau/4$. By Chernoff bound we have the probability of $\sum_{\ell \in [m]} \boldsymbol{\mathcal{X}}_\ell \geq (1 - \tau/2)m$ is $1 - e^{-\Omega(n)}$. We show that when this occurs we must have $|I \cap (\boldsymbol{S}_\ell \setminus \boldsymbol{B}_\ell)| \geq \gamma M$ for some length-M interval I for some $\ell \in [m]$, which implies the same for $\boldsymbol{S}_m \setminus \boldsymbol{B}_m$ given that $\boldsymbol{S}_\ell \subseteq \boldsymbol{S}_m$.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

To this end we assume for a contradiction that this is not the case for any $\ell \in [m]$. Then every $\boldsymbol{\mathcal{X}}_\ell = 1$ implies (3.3). Given that we always have $\boldsymbol{S}_{\ell-1} \subseteq \boldsymbol{S}_\ell$, we have

$$|\boldsymbol{S}_m| \geq ((1-\tau/4)|C|)^{(1-\tau/2)m} \geq |C|^m |C|^{-(\tau/2)m} e^{-(\tau/2)m} \geq M \cdot |C|^{\tau n - \tau m} = M \cdot 2^{\Omega(n)},$$

Here the second step used $1 - \tau/4 \geq e^{-\tau/2}$ for our small τ. The third step used $m = \lceil \tau n + \log_{|C|} M \rceil$ and $e < 3 \leq |C|$. On the other hand, by Claim 1,

$$|\boldsymbol{B}_m| \leq |C|^m \cdot 2|C|^{-2\tau n} < M.$$

As a result, we have $|\boldsymbol{S}_m \setminus \boldsymbol{B}_m| \geq M \cdot 2^{\Omega(n)}$, a contradiction with the trivial upper bound of $\epsilon n M/4$ for $|\boldsymbol{S}_m \setminus \boldsymbol{B}_m|$ by definition. We prove Claim 2 in the rest of the proof:

Proof of Claim 2. Fix an $\ell \in [m]$ and $\vec{y} \in [0 : M-1]^{\ell-1}$ such that $|I \cap (\mathcal{S}(\vec{y}) \setminus \mathcal{B}(\vec{y})| < \gamma M$ for any length-M interval I. Let $\boldsymbol{z} \sim [0 : M-1]$, and we consider the quantity $|C| \cdot |\mathcal{S}(\vec{y})| - |\mathcal{S}(\vec{y} \circ \boldsymbol{z})|$ as a random variable. It is easy to see that this random variable is nonnegative (as each element in $\mathcal{S}(\vec{y})$ gives rise to at most $|C|$ elements in $\mathcal{S}(\vec{y} \circ \boldsymbol{z})$). We will show that

$$\mathop{\mathbf{E}}_{\boldsymbol{z}} \Big[|C| \cdot |\mathcal{S}(\vec{y})| - |\mathcal{S}(\vec{y} \circ \boldsymbol{z})| \Big] \leq |\mathcal{S}(\vec{y})| \cdot \Big(2|C|M \cdot \gamma + |\mathcal{B}(\vec{y})| \Big) \cdot \frac{|C|^2}{M}. \tag{3.4}$$

We show (3.4) by upper-bounding the number of "collisions" that occur when we derive elements in $\mathcal{S}(\vec{y} \circ \boldsymbol{z})$. Namely, consider the pairs (w, c) for $w \in \mathcal{S}(\vec{y})$ and $c \in C$. If every two distinct pairs $(w, c) \neq (w', c')$ were to evaluate to $w + c\boldsymbol{z} \neq w' + c'\boldsymbol{z}$, then there is no *collision* and we would have $|\mathcal{S}(\vec{y} \circ \boldsymbol{z})| = |C| \cdot |\mathcal{S}(\vec{y})|$. In general $|C| \cdot |\mathcal{S}(\vec{y})| - |\mathcal{S}(\vec{y} \circ \boldsymbol{z})|$ is at most the number of collisions.

To bound the number of collisions, we observe that a necessary condition for $w + c\boldsymbol{z} = w' + c'\boldsymbol{z}$ to happen is $|w - w'| \leq 2dM$ since $c, c' \in C = \{0, \pm 1, \ldots, \pm d\}$ and $\boldsymbol{z} \in [0 : M-1]$. Below we bound the number of pairs $(w, c) \neq (w', c')$ such that $|w - w'| \leq 2dM$:

- Clearly, there are at most $|\mathcal{S}(\vec{y})|$ possibilities for w and at most $|C|^2$ possibilities for c and c'.
- By assumption, $\mathcal{S}(\vec{y}) \setminus \mathcal{B}(\vec{y})$ does not occupy a γ fraction of any length-M interval. Hence the interval $[w - 2dM, w + 2dM]$ contains at most $4dM \cdot \gamma$ many elements in $\mathcal{S}(\vec{y}) \setminus \mathcal{B}(\vec{y})$. Moreover, this interval trivially has at most $|\mathcal{B}(\vec{y})|$ many elements in $\mathcal{B}(\vec{y})$. So in total there could be at most $4dM \cdot \gamma + |\mathcal{B}(\vec{y})|$ elements $w' \in \mathcal{S}(\vec{y})$ such that $|w - w'| \leq 2dM$.

As a result, the number of pairs $(w, c) \neq (w', c')$ with $|w - w'| \leq 2dM$ is at most

$$|\mathcal{S}(\vec{y})| \cdot |C|^2 \cdot \Big(2|C|M \cdot \gamma + |\mathcal{B}(\vec{y})| \Big).$$

Then (3.4) follows as each pair leads to a collision with probability at most $1/M$ over $\boldsymbol{z} \sim [0 : M-1]$.

Using Claim 1 and our choice of m, we have

$$|\mathcal{B}(\vec{y})| \leq |C|^m \cdot 2|C|^{-2\tau n} \leq M \cdot 2|C|^{-\tau n} < |C|M \cdot \gamma. \tag{3.5}$$

Combining (3.4) and (3.5) together (and using our choice of γ in (3.2)), we have

$$\mathop{\mathbf{E}}_{\boldsymbol{z}} \left[|C| - \frac{|\mathcal{S}(\vec{y} \circ \boldsymbol{z})|}{|\mathcal{S}(\vec{y})|} \right] \leq \Big(2|C|M \cdot \gamma + |\mathcal{B}(\vec{y})| \Big) \cdot \frac{|C|^2}{M} \leq 3|C|^3 \cdot \gamma = \frac{|C| \cdot \tau^2}{16}.$$

Given that the random variable in the expectation is nonnegative, we have from Markov that

$$\mathop{\mathbf{Pr}}_{\boldsymbol{z}} \left[\frac{|\mathcal{S}(\vec{y} \circ \boldsymbol{z})|}{|\mathcal{S}(\vec{y})|} < (1 - \tau/4)|C| \right] \leq \frac{\tau}{4}.$$

The claim follows. □

This finishes the proof of the Dense Interval Lemma. □

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Next we prove the lemma for Step 2:

Lemma 2 (Interval Shifting Lemma)**.** *Let $\vec{x} \in [0 : M-1]^m$ be such that $\mathcal{S}(\vec{x}) \setminus \mathcal{B}(\vec{x})$ occupies at least γ-fraction of some length-M interval. Let $\vec{\boldsymbol{y}} \sim [0 : M-1]^{m'-m}$. Then with probability at least $1 - e^{-\Omega(n)}$, $\mathcal{S}(\vec{x} \circ \vec{\boldsymbol{y}})$ occupies γ-fraction of some length-M interval that contains τ.*

Proof. Let $I = [\alpha, \alpha + M] \subseteq [-\epsilon nM/8, \epsilon nM/8]$ be a length-M interval such that $J := I \cap (\mathcal{S}(\vec{x}) \setminus \mathcal{B}(\vec{x}))$ satisfies $|J| \geq \gamma M$. Assume without loss of generality that $\tau \notin I$ and $\tau < \alpha$. As $|\tau| = o(Mn)$, we have $0 < \alpha - \tau < \epsilon nM/7$. For $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{m'-m} \sim [0 : M-1]$, it follows from Hoeffding inequality (note that the sum has expectation $(m'-m)(M-1)/2 \geq (M-1) \cdot \epsilon n/6$ as $m' - m \geq \epsilon n/3$) that

$$\sum_{i \in [m'-m]} \boldsymbol{y}_i \geq \epsilon nM/7 > \alpha - \tau.$$

When this happens, we let ℓ be the smallest index such that

$$\beta := \sum_{i \in [\ell]} \boldsymbol{y}_i > \alpha - \tau$$

so $\beta - M \leq \alpha - \tau$. On the one hand $\mathcal{S}(\vec{x} \circ \vec{\boldsymbol{y}})$ occupies at least γ-fraction of $[\alpha - \beta, \alpha - \beta + M]$ since it contains J after shifting down by β. On the other hand, we have $\tau > \alpha - \beta$ and $\tau \leq \alpha - \beta + M$. This finishes the proof of the lemma. □

Now we are ready to conclude the proof of Theorem 4. Combining Lemma 1 and Lemma 2 we have with probability at least $1 - e^{-\Omega(n)}$ over $\vec{\boldsymbol{x}} \sim [0 : M-1]^{m'}$, $\mathcal{S}(\vec{\boldsymbol{x}})$ contains at least γ-fraction of a length-M interval that contains τ. Fix such a $\vec{x} \in [0 : M-1]^{m'}$. We draw the next $n - m' = \Omega(n)$ random elements $\boldsymbol{x}_{m'+1}, \ldots, \boldsymbol{x}_n \sim [0 : M-1]$. For each such $\boldsymbol{x}_i$, there is at least a γ probability that either $w + \boldsymbol{x}_i = \tau$ or $w - \boldsymbol{x}_i = \tau$ for some $w \in \mathcal{S}(\vec{x})$. When this happens for some i, we get a solution by setting coefficients $c_1, \ldots, c_{m'}$ to achieve w in the sum over $\vec{x}$, setting c_i to be either 1 or -1 accordingly, and setting every other $c_{m'+1}, \ldots, c_n$ to be 0. The probability that all $n - m'$ such $\boldsymbol{x}_i$'s fail to yield a solution is at most $(1-\gamma)^{\Omega(n)} = e^{-\Omega(n)}$. (Note that the same proof works for the case when $\tau = 0$ since during Step 3, we always get a not-all-zero solution with at least one $\{\pm 1\}$-coefficient when the event happens for some $\boldsymbol{x}_i$.)

Taking a union bound over the failure probability at each of the three steps, we get that there is a solution with probability $1 - e^{-\Omega(n)}$ when $M \leq |C|^{(1-\epsilon)n}$. This finishes the proof of Theorem 4.

4 Algorithmic Results

This section proves Theorem 1. Using structural results from the previous section, we start by showing that it suffices to design an algorithm for the core case when $M \in |C|^{(1\pm\epsilon)n}$ for a small positive constant ϵ.

4.1 Reduction to the Core Problem

We start with some notation. Given $C = C(d)$ or $C_0(d)$ for some fixed $d \in \mathbb{N}_{\geq 1}$, a *solution profile* $\pi = (\pi_w)_{w \in C}$ is a tuple of nonnegative integers that sum to n. For an input vector $\vec{x} = (x_1, \ldots, x_n)$, a target offset τ and a (solution) profile π, we say $\vec{c} \in C^n$ is a solution to GSS under the profile π if $\vec{c} \cdot \vec{x} = \tau$ and the number of occurrences of w in vector $\vec{c}$ is π_w for each coefficient $w \in C$.

Our goal of this section is to prove the following theorem for the problem of solving GSS under a given profile:

Theorem 5. *Fix any $d \in \mathbb{N}_{\geq 1}$ and let $C = C(d)$ or $C_0(d)$. For any sufficiently small constant $\xi > 0$, there is a constant $\epsilon > 0$ and a randomized algorithm with running time $|C|^{\Lambda(|C|)n + \xi n}$ that has the following performance guarantee. Given any $M \in |C|^{(1\pm\epsilon)n}$, τ with $|\tau| = o(nM)$ and a profile π, the algorithm succeeds[8] on $(M, \tau, \pi, \vec{\boldsymbol{x}})$ with probability at least $1 - e^{-\Omega(n)}$ (over the draw of $\vec{\boldsymbol{x}} \sim [0 : M-1]^n$ and the randomness of the algorithm).*

We first use Theorem 5 and our structural results to prove Theorem 1; the proof of Theorem 2 is similar and can be found in Appendix B. The intuition is that if $M \geq |C|^{(1+\epsilon)n}$ then most likely there is no solution but if $M \leq |C|^{(1-\epsilon)n}$, then we can reduce the instance size to n' so that M falls inside the window $|C|^{(1\pm\epsilon)n'}$ and the algorithm of Theorem 5 applies.

[8] Similar to the original GSS problem, an algorithm fails if it returns "no solution" but indeed there is a solution under the given profile π.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Proof of Theorem 1 assuming Theorem 5. Fix a constant $\zeta > 0$ as in the statement of Theorem 1. We will show an algorithm for average-case GSS with running time $|C|^{\Lambda(|C|)n+\zeta n}$, assuming Theorem 5. Set $\xi \le \zeta/2$ to be sufficiently small that Theorem 5 holds, and let $\epsilon > 0$ be the constant determined by ξ in Theorem 5. We begin by considering $C = C_0(d)$ and split into three cases based on the size of M.

First, if $M \ge |C|^{(1+\epsilon)n}$ then it follows from Theorem 4 that the probability of having a solution is $e^{-\Omega(n)}$ and thus returning "no solution" achieves overall success probability $1 - e^{-\Omega(n)}$. Next, if $M \in |C|^{(1\pm\epsilon)n}$ then we run the algorithm in Theorem 5 for every profile π and return any solution it finds; we return "no solution" if no solution is found for any profile π. Given that there are only polynomially many profiles (since d is fixed), the success probability remains $1 - e^{-\Omega(n)}$ by a union bound and the running time only goes up by a polynomial factor.

Finally we consider the case when $M \le |C|^{(1-\epsilon)n}$ but $M = |C|^{\Omega(n)}$. Let $n' = \Omega(n)$ be an integer such that M is between $|C|^{(1-\epsilon)n'}$ and $|C|^{(1-\epsilon/2)n'}$, and note that τ still satisfies $|\tau| = o(n'M)$ given $n' = \Omega(n)$. We run the algorithm in Theorem 5 on the first n' input integers $\vec{\boldsymbol{x}}' = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_{n'})$ and try all possible profiles. It follows from Theorem 4 that there is a solution with probability at least $1 - e^{-\Omega(n')} = 1 - e^{-\Omega(n)}$ and thus a solution will be found by the algorithm of Theorem 5 with probability $1 - e^{-\Omega(n)}$. Given that $C = C_0(d)$, any solution to $\vec{\boldsymbol{x}}'$ can be extended to obtain a solution to $\vec{\boldsymbol{x}}$ by assigning the 0 coefficient to the remaining inputs. So in this case our algorithm finds a solution to $\vec{\boldsymbol{x}}$ with probability $1 - e^{-\Omega(n)}$.

Second, consider $C = C(d)$. The first two cases, $M \ge |C|^{(1+\epsilon)n}$ and $M \in |C|^{(1\pm\epsilon)n}$, are similar, except that in the second case by answering "no solution" we achieve only a success probability of $1 - o_n(1)$ by Corollary 2 and Theorem 3; the success probability in the first case remains $1 - e^{-\Omega(n)}$. The main difference occurs in the last case, as we can no longer extend a solution to $\vec{\boldsymbol{x}}'$ to a solution to $\vec{\boldsymbol{x}}$ by setting the coefficients of every other input integer to be 0. Instead, we can shrink the input instance as follows. Reveal the last $n - n'$ input integers one by one. For each input $\boldsymbol{x}_i$, $i \in [n' + 1 : n]$, perform the following operation: if the current target τ is positive, assign -1 to $\boldsymbol{x}_i$ and subtract it from τ to create a new target. If the current target τ is negative, assign $+1$ to $\boldsymbol{x}_i$ and add it to τ to create a new target. After this procedure, we are left with n' random numbers $\vec{\boldsymbol{x}}' = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_{n'})$ and a new offset τ' with $|\tau'| = o(n'M)$ such that any solution to $(\vec{\boldsymbol{x}}, \tau')$ can be extended to a solution to $(\vec{\boldsymbol{x}}, \tau)$. By Theorem 3, there is a solution with probability at least $1 - o_n(1)$ over the randomness of $\vec{\boldsymbol{x}}'$ in the $C = C(d)$, $d > 1$ case. By Corollary 2, there is a solution with probability at least $1 - o_n(1)$ over the randomness of $\vec{\boldsymbol{x}}'$ in the $C = C(1)$ case if $\sum_i \boldsymbol{x}_i$ has the same parity as τ, as the shrinking procedure preserves the parity of $\sum_i \boldsymbol{x}_i - \tau$. This implies that running the algorithm of Theorem 5 on $\vec{\boldsymbol{x}}'$ and τ' for all profiles π finds a solution with probability at least $1 - o_n(1)$, which can be extended to a solution to $\vec{\boldsymbol{x}}$ and τ. □

4.2 Algorithm Overview We give an overview of our algorithmic approach for Theorem 5 and the underlying definitions and ingredients required for the proof. For the rest of the section, we consider d to be a fixed integer with $C = C(d)$ or $C_0(d)$. $\delta > 0$ will denote a sufficiently small constant and $\epsilon := \epsilon(\delta) > 0$ a smaller constant that depends on δ, defined later in the proof. Our goal is to give an algorithm for GSS with a solution profile π that runs in time $|C|^{\Lambda(|C|)n+O(\delta n)}$ and has success probability $1 - e^{-\Omega(n)}$ when $M \in |C|^{(1\pm\epsilon)}$ and $|\tau| = o(nM)$.

4.2.1 Terminology and Key Notions For the rest of the proof, the coefficient set $D := D(z)$ denotes a translation of the coefficient set C by some *nonzero* $z \in C$, i.e., $D = \{w - z : w \in C\}$. Note that we always have $0 \in D$ whether $C = C_0(d)$ or $C(d)$.

Given $D = D(z)$ for a certain nonzero $z \in C$, we define *input partitions* and *half partitions*:

Definition 1. A *solution profile* with respect to D is a tuple of nonnegative integers $(\sigma_w)_{w\in D}$ that sum to n. An *input partition* of $[n]$ with respect to D is a tuple of pairwise disjoint sets $\mathbb{S} := (S_w)_{w\in D}$ with union $[n]$. We say $\mathbb{S}$ corresponds to the solution profile σ if $|S_w| = \sigma_w$ for all $w \in D$. The *size* of a solution profile σ is given by $|\sigma| := n - \sigma_0$; similarly the *size* of an input partition $\mathbb{S}$ is given by $|\mathbb{S}| := \sum_{w\in D\setminus\{0\}} |S_w|$.

Definition 2 (Half-partitions)**.** Given a solution profile σ with respect to D, a *half-partition* that corresponds to σ is a tuple of sets $\mathbb{T} := (T_w)_{w\in D\setminus\{0\}}$ that satisfies the following conditions:

1. $|T_w| = \sigma_w/2$ for $w \in D \setminus \{0\}$ if σ_w is even.

2. $|T_w| = (\sigma_w + 1)/2$ or $(\sigma_w - 1)/2$ for $w \in D \setminus \{0\}$ if σ_w is odd.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Given an input partition $\mathbb{S}$ that corresponds to σ, we say $\mathbb{T}$ is a half-partition of $\mathbb{S}$ if $T_w \subseteq S_w$ for all $w \in D \setminus \{0\}$. Two half-partitions $\mathbb{T} = (T_w)_{w \in D \setminus \{0\}}$ and $\mathbb{T}' := (T'_w)_{w \in D \setminus \{0\}}$ of $\mathbb{S}$ are called a *matching pair* of $\mathbb{S}$ if for every $w \in D \setminus \{0\}$, T_w and T'_w are disjoint and their union is S_w.

Given a half-partition $\mathbb{T} = (T_w)_{w \in D \setminus \{0\}}$ and an input partition $\mathbb{S} = (S_w)_{w \in D}$, their *signatures* under the input vector $\vec{x}$ are defined as

$$\operatorname{sig}(\mathbb{T}, \vec{x}) := \sum_{w \in D \setminus \{0\}} w \cdot \Big(\sum_{i \in T_w} x_i \Big) \quad \text{and} \quad \operatorname{sig}(\mathbb{S}, \vec{x}) := \sum_{w \in D \setminus \{0\}} w \cdot \Big(\sum_{i \in S_w} x_i \Big).$$

For a fixed input partition $\mathbb{S}$, we let $A(\mathbb{S})$ denote the set of all half-partitions of $\mathbb{S}$. Let $B(\sigma)$ denote the set of all half-partitions that correspond to σ. For brevity, we write $a := a(\sigma) = |A(\mathbb{S})|$, as a is the same for all $\mathbb{S}$ corresponding to σ, and $b := b(\sigma) = |B(\sigma)|$ when σ is clear from context.

4.2.2 High-Level Algorithm Sketch Let π be the input solution profile with respect to $C = C_0(d)$ or $C(d)$. Our algorithm starts by picking a nonzero $z \in C$ and translating the coefficient set to $D = C - z$. We can use π to induce a solution profile σ with respect to D, where $\sigma_w = \pi_{w+z}$ for each $w \in D$. Our goal is now to find an input partition $\mathbb{S}$ (with respect to D) that corresponds to σ such that

$$\operatorname{sig}(\mathbb{S}, \vec{x}) = \tau - z \cdot \sum_{i \in [n]} x_i. \tag{4.6}$$

We set $z = \max_{z' \in C \setminus \{0\}} \pi_{z'}$, in order to minimize the size $|\sigma|$ of solution profile σ. This is natural because it allows us to focus on small-size candidate solutions (input partitions) after the translation. When $C = C(d)$, there exists a $z \in C$ such that the translated profile σ has size $|\sigma| \le \frac{|C|-1}{|C|} n$ by the pigeonhole principle. And when $C = C_0(d)$, this is not possible in general because π_0 can be large. (Looking ahead, we require $z \neq 0$ for our proof of Lemma 4). To this end, we prove the following technical lemma, which states that if $\pi_0 \ge (\frac{1}{|C|} + \delta)n$ then there is no solution with probability at least $1 - e^{-\Omega(n)}$. This allows to focus on the case when $\pi_0 < (\frac{1}{|C|} + \delta)n$, in which case we can choose $z \in C$ such that σ satisfies

$$|\sigma| \le \left(\frac{|C| - 1}{|C|} + \frac{\delta}{|C| - 1} \right) n. \tag{4.7}$$

Lemma 3. *Suppose $M \in |C|^{(1 \pm \epsilon)n}$ for some sufficiently small constant $\epsilon = \epsilon(\delta) > 0$. If $\pi_0 \ge (\frac{1}{|C|} + \delta)n$, then $\vec{\boldsymbol{x}} \sim [0 : M - 1]^n$ has no solution with probability at least $1 - e^{-\Omega(n)}$.*

Proof. This follows from taking a union bound over all solutions corresponding to π. When $\pi_0 \ge (\frac{1}{|C|} + \delta)n$, the number of candidate solutions is at most $O^*(2^{hn})$ where

$$h := H\left(\frac{1}{|C|} + \delta, \frac{1}{|C|} - \frac{\delta}{|C| - 1}, \ldots, \frac{1}{|C|} - \frac{\delta}{|C| - 1} \right) = \log_2 |C| - f(\delta),$$

where f is a nonnegative, increasing function on $[0, 1]$. It follows from holding out one input element corresponding to a nonzero coefficient that each candidate solution is a solution with probability at most $1/M$. Taking the union over all candidate solutions, we have that for $\epsilon < f(\delta)$, the probability of having a solution under the profile π is $e^{-\Omega(n)}$. □

From now on, we consider the translated coefficient set D and solution profile σ that satisfy the size bound in (4.7). (Notice that $|D| = |C|$.) Our goal is to find an input partition that satisfies (4.6) when it exists. However, even with its relatively small size, the profile σ typically still corresponds to an exponential number of input partitions. Assume that $\mathbb{S}$ is an input partition satisfying (4.6). At a high level, our algorithm searches for $\mathbb{S}$ by exploiting the fact that every input partition $\mathbb{S}$ with respect to σ can be decomposed into a matching pair of half-partitions $(\mathbb{T}, \mathbb{T}')$ in many ways. Lemma 4, which we refer to as the *Signature Distribution Lemma*, shows that with high probability over $\vec{\boldsymbol{x}} \sim [0 : M - 1]^n$, most half-partitions of $\mathbb{S}$ have distinct signatures. Assuming this event occurs, we need to recover one matching pair of half-partitions in $A(\mathbb{S})$ from the larger search space

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Algorithm 1: Skeleton Algorithm for GSS

Constants: $C = C(d)$ or $C_0(d)$, sufficiently small $\delta > 0$ and smaller $\epsilon = \epsilon(\delta) > 0$
Input: $M \in |C|^{(1\pm\epsilon)n}$, τ, $\vec{x} \in [0 : M-1]^n$, a profile σ of $D = C - z$ for some nonzero $z \in C$.
Define $a := a(\sigma)$, $b := b(\sigma)$ as in (4.8) and (4.9)
Define $P := \min(b^{2/3}a^{-1/3}, a/n)$
repeat poly(n) **times**
  Select a prime $\boldsymbol{p} \in [P, 2P]$ and an integer $\boldsymbol{r} \in [0 : p-1]$ uniformly at random.
  If $P = a/n$, let $\boldsymbol{L}_1 = \{\mathbb{T} \in B(\sigma) \mid \mathrm{sig}(\mathbb{T}, \vec{x}) = \boldsymbol{r} \pmod{\boldsymbol{p}}\}$. If $P = b^{2/3}a^{-1/3}$, let $\boldsymbol{L}_1$ be a subsample of $O^*(b/\sqrt{a\boldsymbol{p}})$ elements drawn from this set with replacement.
  Let $\boldsymbol{L}_2$ be defined as $\boldsymbol{L}_1$ with respect to $\{\mathbb{T} \in B(\sigma) \mid \mathrm{sig}(\mathbb{T}, \vec{x}) = \tau - z\sum_i x_i - \boldsymbol{r} \pmod{\boldsymbol{p}}\}$.
  Sort $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$ by signature and use Meet-in-the-Middle to find a disjoint pair $(\mathbb{T}_1, \mathbb{T}_2)$ satisfying $\mathrm{sig}(\mathbb{T}_1, \vec{x}) + \mathrm{sig}(\mathbb{T}_2, \vec{x}) = \tau - z\sum_i x_i$ if one exists.
end

$B(\sigma)$. (Note that our bound on the size of σ comes into play at this point: a profile σ of smaller size would imply a larger ratio a/b and make our search easier.)

To recover a solution, we exploit the fact that the signatures of any matching pair sum to $\tau - z\sum_i x_i$ but at the same time, by the Signature Distribution Lemma, most half-partitions in $A(\mathbb{S})$ have distinct signatures. We (virtually) hash the half-partitions based on signature by grouping them into residue classes (which we will refer to as buckets) modulo a large random prime $\boldsymbol{p}$ with a carefully picked magnitude. After the hash, most buckets contain a small fraction of the search space. We then sample pairs of buckets whose elements sum to $\tau - z\sum_i x_i \pmod{\boldsymbol{p}}$ and either subsample from or exhaustively search over the paired buckets to recover a solution.

Algorithm 1 outlines the algorithm in pseudocode. Certain implementation details are deferred to the proofs of correctness (Proposition 1) and runtime (Proposition 2) to simplify the presentation.

As mentioned earlier, we prove a Signature Distribution Lemma (Lemma 4) in Section 4.3 that shows that when the instance has a solution, the corresponding input partition almost always decomposes into many pairs of half-partitions with distinct signatures. (If this condition is false, most matching pairs of half-partitions hash to the same buckets and the sampling procedure fails.)

It remains to choose a value for the endogenous parameter P that determines the magnitude of our modulus $\boldsymbol{p}$. Recall that $a := a(\sigma) = |A(\mathbb{S})|$ counts the number of half-partitions of any input partition $\mathbb{S}$ corresponding to σ. We observe that

$$a(\sigma) = \Theta^*(2^{|\sigma|}). \tag{4.8}$$

Likewise, recall that $b := b(\sigma)$ denotes the maximum number of half-partitions corresponding to σ. We have $b(\sigma) = O^*(2^{hn})$ where (writing $|\sigma| = \alpha n$)

$$h := H\left(\frac{\alpha}{2(|C|-1)}, \ldots, \frac{\alpha}{2(|C|-1)}, 1 - \frac{\alpha}{2}\right) = \frac{\alpha}{2} \cdot \log_2 \frac{2(|C|-1)}{\alpha} + \left(1 - \frac{\alpha}{2}\right) \cdot \log_2 \frac{2}{2-\alpha}. \tag{4.9}$$

We prove in Section 4.3 that almost all yes-instances have solutions that decompose into half-partitions with $\Omega(a)$ distinct signatures. In this case, we expect most residue classes to contain $\Omega(a/\boldsymbol{p})$ half-partitions that are part of matching pairs and $O(b/\boldsymbol{p})$ half-partitions in total. Setting $P = a/n$ thus ensures that every pair of residue classes $\boldsymbol{r} \pmod{\boldsymbol{p}}$ and $\tau - z\sum_i x_i\boldsymbol{r} \pmod{\boldsymbol{p}}$ contains $\Omega(n)$ matching pairs in expectation. By repeatedly sampling bucket pairs and using the Meet-in-the-Middle algorithm to search for a matching pair, we recover a solution.

In fact, for $|C| \le 3$, including the case $C = C_0(1) = \{0, \pm 1\}$ that is our primary interest, it is possible to implement a slightly more efficient sampling procedure. In these cases, we set $P = b^{2/3}a^{-1/3} < a$, so that each residue class pair contains an exponential number of matching pairs in expectation. After choosing a bucket pair, we then subsample each bucket to create solution lists that are likely to contain at least one matching pair by a birthday paradox argument. We then use the Meet-in-the-Middle algorithm on the subsampled lists to recover a solution as before.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

4.2.3 Section Outline In the next subsection we prove the Signature Distribution Lemma. We prove the correctness of the algorithm in Proposition 1, where we show that for any $\vec{x}$ that satisfies the conditions of the Signature Distribution Lemma, Algorithm 1 recovers $\mathbb{S}$ with high probability. Finally, the proof of Proposition 2 analyzes the runtime of our algorithm and fills in several implementation details omitted from the pseudocode.

4.3 Signature Distribution Lemma For our approach to work, we need the half-partitions of some solution $\mathbb{S}$ to have many distinct signatures. While we do not know how to guarantee this for worst-case inputs, the following lemma gives what we need for average-case inputs. Essentially, it says that with very high probability, a random instance $\vec{\boldsymbol{x}} \sim [0 : M-1]^n$ either has no solutions or has a solution $\mathbb{S}$ such that different half-partitions have many different signatures.

For $C = C_0(1)$, there is a combinatorial argument that shows that signatures are well-distributed in the worst case (see [MNPW19] for details).[9] We employ a probabilistic argument that holds for general C.

Lemma 4 (Signature Distribution Lemma)**.** *Let $C = C_0(d)$ or $C(d)$ and $M \in |C|^{(1\pm\epsilon)n}$ for some constant $\epsilon > 0$. Let σ be a solution profile with respect to $D = C - z$ for some $z \in C \setminus \{0\}$ with $|\sigma|$ satisfying*

$$|\sigma| \le \Big(\frac{|C|-1}{|C|} + \frac{\delta}{|C|-1}\Big)n.$$

With probability $1 - e^{-\Omega(n)}$, $\vec{\boldsymbol{x}} \sim [0 : M-1]^n$ either has no solution $\mathbb{S}$ to $\operatorname{sig}(\mathbb{S}, \vec{\boldsymbol{x}}) = \tau - z\sum_i \boldsymbol{x}_i$, or there is a solution $\mathbb{S}$ such that half-partitions of $\mathbb{S}$ have at least $\Omega(a(\sigma))$ many distinct signatures.

Proof. Let $a := a(\sigma)$ as in (4.8). For the lemma to fail, there must be an input partition $\mathbb{S} = (S_w)_{w\in D}$ that corresponds to σ such that (1) $\operatorname{sig}(\mathbb{S}, \vec{\boldsymbol{x}}) = \tau - z\sum_i \boldsymbol{x}_i$ and (2) half-partitions of $\mathbb{S}$ have $o_n(a)$ distinct signatures under $\vec{x}$. Now fix any input partition $\mathbb{S}$ that corresponds to σ. We will bound the number of $\vec{x} \in [0 : M-1]^n$ such that both of the conditions above hold by $O(aM^{n-2})$. Since there are no more than $|C|^n$ many input partitions $\mathbb{S}$, the number of such $\vec{x}$ is thus at most

$$\begin{aligned} |C|^n \cdot O(aM^{n-2}) &= O^*(M^n \cdot |C|^{2\epsilon n} \cdot 2^{(-\frac{1}{|C|} + \frac{\delta}{|C|-1})n}) \\ &\le O^*(M^n \cdot 2^{-(\frac{1}{|C|} - \frac{\delta}{|C|-1} - 2\epsilon \log_2 |C|)n}), \end{aligned}$$

given that $a = \Theta^*(2^{|\sigma|}) = O^*(2^{(1-\frac{1}{|C|}+\frac{\delta}{|C|-1})n})$ and $2^n \le |C|^n \le M \cdot |C|^{\epsilon n}$. This is an exponentially small fraction of the input space, which has size M^n, when both ϵ and δ are sufficiently small.

Fixing an input partition $\mathbb{S} = (S_w)_{w\in D}$, we proceed to bound the number of $\vec{x}$ that satisfy both conditions above. We start by showing that the number of $\vec{x}$ such that half-partitions of $\mathbb{S}$ have $o(a)$ distinct signatures is at most $O(aM^{n-1})$.

To see this, consider any of the a^2 pairs of distinct half-partitions of $\mathbb{S}$. With respect to any pair of distinct half-partitions, there exists an index $k \in [n]$ that appears in exactly one of them. For any fixed $\vec{x}_{[n]\setminus\{k\}}$, there is at most one choice of x_k that results in both half-partitions having the same signature. As a result, the number of $\vec{x}$ that can lead to the same signature is at most M^{n-1}. Union-bounding the number of signature-collisions over a^2 distinct pairs of half-partitions of $\mathbb{S}$ and M^n instances gives at most a^2M^{n-1} signature-collisions in total over all instances $\vec{x}$. This implies that the number of $\vec{x}$ that have $o(a)$ distinct half-partition signatures, which happens only if there occur $\Omega(a)$ signature-collisions, is at most $O(aM^{n-1})$.

Next we observe that the condition of $\vec{x}$ corresponding to $o(a)$ distinct half-partition signatures is independent of $\vec{x}_{S_0}$, as these elements do not affect half-partition signatures at all. On the other hand, $\vec{x}_{S_0}$ determines whether the equation

$$\operatorname{sig}(\mathbb{S}, \vec{x}) = \tau - z\sum_{i\in[n]} x_i \tag{4.10}$$

holds because its entries appear on the right hand side with a nonzero coefficient z. By holding out any input element indexed by S_0, we can see that among the $O(aM^{n-1})$ many input vectors $\vec{x}$ that have $o(a)$ distinct

[9] In fact, this argument can be generalized to $C_0(d)$. However, it relies crucially on the existence of the 0 coefficient and a coefficient set that is symmetric about 0, and thus cannot be easily extended to $C(d)$ or translated instances.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

half-partition signatures, only $1/M$-fraction of them can satisfy (4.10). Therefore the number of $\vec{x}$ that satisfies both conditions is at most $O(aM^{n-2})$.

This finishes the proof of the lemma. □

4.4 Correctness and Runtime

Proposition 1 (Correctness of Algorithm 1)**.** *Under the setting of Theorem 5, Algorithm 1 succeeds on* $(M, \tau, \pi, \vec{\boldsymbol{x}})$ *with probability at least* $1 - e^{-\Omega(n)}$.

Proof. By the Signature Distribution Lemma, we may assume that either $\vec{x}$ has no solution or there is an input partition $\mathbb{S}$ such that $\mathrm{sig}(\mathbb{S}, \vec{x}) = \tau - z\sum_{i\in[n]} x_i$ and half-partitions of $\mathbb{S}$ have $\Omega(a)$ many distinct signatures. We assume the latter case because Algorithm 1 never returns false solutions.

We proceed to argue that for most primes $p \in [P, 2P]$, the signatures of $A(\mathbb{S})$ are distributed over many residue classes modulo p. To see this, consider any two distinct signatures s_1, s_2 of elements of $A(\mathbb{S})$. Because $|s_1 - s_2| < n|C|M = 2^{O(n)}$ and $P = 2^{\Omega(n)}$, the value $|s_1 - s_2|$ is divisible by $O_n(1)$ prime factors larger than P. Moreover, for sufficiently large P, there exist at least $P/\log(P)$ prime integers in the interval $[P, 2P]$. (This follows from the Prime Number Theorem. See, e.g., [MNPW19, Lemma 2.1].) Thus for a prime $\boldsymbol{p}$ sampled uniformly at random from all primes in the interval $[P, 2P]$ we have

$$\Pr_{\boldsymbol{p}}\left[\boldsymbol{p} \text{ divides } |s_1 - s_2|\right] = O(\log(P)/P).$$

Let $a' = \Omega(a)$ be the number of distinct signatures of $A(\mathbb{S})$. The expected number of pairs of half-partitions whose signatures collide mod $\boldsymbol{p}$ is $O(a'^2 \log(P)/P)$ by linearity of expectation. By Markov's inequality, there exists a constant ρ_1 such that a randomly selected prime $\boldsymbol{p} \in [P, 2P]$ corresponds to at most $\rho_1 a'^2 \log(P)/P$ signature-collisions with probability at least $1/2$. Call such a prime $\boldsymbol{p} \in [P, 2P]$ with at most $\rho_1 a'^2 \log(P)/P$ signature-collisions *a good prime* $\boldsymbol{p}$.

It remains to show that for a good prime p, choosing a uniform random residue $\boldsymbol{r} \in [p]$ will allow us to recover $\mathbb{S}$ with inverse polynomial probability. With respect to a good prime p, we define a *good residue* $\boldsymbol{r}$ to be one that satisfies the following two conditions:

1. The signatures of at least $a'/2p$ half-partitions in $A(\mathbb{S})$ are equal to $\boldsymbol{r} \pmod p$.
2. The signatures of at most $\rho_2 bn^2/p$ half-partitions in $B(\sigma)$ are equal to either $\boldsymbol{r} \pmod p$ or $\tau - \boldsymbol{r} - z\sum_i x_i \pmod p$, where $b := |B(\sigma)|$, for a constant ρ_2 defined below.

Fix a good prime p. We first prove that $\Omega(p/n^2)$ residue classes each contain at least $a'/2p$ half-partition signatures corresponding to $\mathbb{S}$. To see this, assume for contradiction that at most p/n^2 residue classes contain at least $a'/2p$ solution half-partitions (we call such residue classes "large"). Under this assumption, residue classes that are not large contain fewer than $a'/2p$ solution half-partitions, so at least $a'/2$ half-partitions must fall into large residue classes. However, reasoning from the case in which signatures are evenly distributed among large residue classes, this implies $\Omega(a'^2 n^2/p)$ signature-collisions, which contradicts the assumption that p is good.

Consider the $\lceil p/2 \rceil$ pairs of residue classes defined by $(r, \tau - r - z\sum_i x_i)$ for some residue $r \in [p]$. Upon choosing a good p, Markov's inequality guarantees that most of the $\Omega(p/n^2)$ residue class pairs that contain $a'/2p$ matching pairs contain $O(bn^2/p)$ half-partitions in total. Thus there exists a constant ρ_2 such that $1/2$ of these residue class pairs contain $\rho_2 bn^2/p$ half-partitions in total. In other words, the chance that uniform random $\boldsymbol{p}$, $\boldsymbol{r}$ are both good is $\Omega(n^{-2})$. We can inflate this probability to $1 - e^{-\Omega(n)}$ by choosing independent pairs $(\boldsymbol{p}, \boldsymbol{r})$ a polynomial number of times as is done in Algorithm 1.

We conclude by proving that Algorithm 1 recovers a solution with very high probability conditioned on the choice of a good p and r. First, consider the case in which $a/n < b^{2/3}a^{-1/3}$, in which case the algorithm sets $P = a/n$. Thus the residue class pair defined by $(r, \tau - r - z\sum_i x_i)$ contains $a'/2p = \Omega(a'n/a) = \Omega(n)$ matching pairs; searching with Meet-in-the-Middle guarantees recovery.

Second, consider the case in which $a/n > b^{2/3}a^{-1/3}$, in which case $P = b^{2/3}a^{-1/3}$. Let

$$\kappa > \frac{a'}{2p} = \frac{a'}{2} \cdot \frac{a^{1/3}}{b^{2/3}} = \Omega\left(\frac{a^{4/3}}{b^{2/3}}\right) = \Omega(n)$$

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

denote the number of matching pairs in the residue class pair defined by $(r, \tau - r - z\sum_i x_i)$. In this case, we create $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$ by subsampling

$$2\rho_2 bn^3/\sqrt{ap} = O^*(b/\sqrt{ap})$$

elements from each residue class uniformly at random with replacement.

In expectation, the number of matching pairs contained in $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$ is thus at least

$$\frac{2\rho_2 bn^3}{\sqrt{ap}} \cdot \frac{\kappa p}{\rho_2 bn^2} \geq 2n\kappa\sqrt{\frac{p}{a}}.$$

Applying a Chernoff bound implies that both lists contain at least $n\kappa\sqrt{\frac{p}{a}} = \Omega(n^{3/2})$ half-partitions that are members of some matching pair with probability $1 - e^{-\Omega(n)}$ as $a/n > p$ by definition.

In other words, with very high probability $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$ each contain

$$n\kappa\sqrt{\frac{p}{a}} \geq n\frac{a'}{2p}\sqrt{\frac{p}{a}} = \Omega(n\sqrt{\kappa})$$

uniform random elements sampled from matching pair sets of size κ. This yields a matching pair in our subsample with probability $1 - e^{-\Omega(n)}$ by the birthday paradox. Searching with Meet-in-the-Middle guarantees recovery. □

Proposition 2 (Runtime of Algorithm 1)**.** *When fully specified as below, Algorithm 1 runs in time* $|C|^{\Lambda(|C|)n+O(\delta n)}$*, where* $\Lambda(z)$ *is defined as in Theorem 1.*

Proof. For fixed prime p and residue class r, we start by describing how to generate $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$ efficiently.

To sample half-partitions by the residue class of their signature, we construct a $n \times p$ dynamic programming table in which each cell $(i, j) \in [n] \times [p]$ stores the number of coefficient vectors $\vec{c} \in D^i$ such that $\vec{c} \cdot x_{[i]} = j \pmod p$. This number is stored as $\text{poly}(n)$ different values indicating how many coefficient vectors match each partial solution profile $\pi = (\pi_w)_{w \in D}$ that partitions the integer i. Each cell (i, j) can be filled by consulting the cells $(i - 1, j - wx_i \pmod p)$ for each $w \in D$. As a result, constructing the table takes time $O^*(p)$.

Our dynamic programming table allows us to sample uniformly at random from the set of half-partitions that correspond to σ and have signatures that fall into the residue class $r \pmod p$. To do this, we begin at cell (n, r) and consider only those coefficient vectors indexed by solution profiles corresponding to half-partitions. We sample a coefficient $w \in D$, weighting by the number of half-partitions that assign the coefficient w to x_n. We then consider cell $(n - 1, r - wx_n \pmod p)$ and continue sampling until a single half-partition is recovered.

Once the table is created, we consider two cases based on the choice of P.

- **Case 1:** $P = a/n$**.** In this case, we sample the entire residue class:

$$\boldsymbol{L}_1 = \{\mathbb{T} \in B(\sigma) \mid \text{sig}(\mathbb{T}, \vec{x}) = r \pmod p\} \quad \text{and}$$
$$\boldsymbol{L}_2 = \{\mathbb{T} \in B(\sigma) \mid \text{sig}(\mathbb{T}, \vec{x}) = \tau - r - z\textstyle\sum_i x_i \pmod p\}.$$

  Under the assumption that p is a good prime and r is a good residue class, this takes time $\text{poly}(n) \cdot \rho_2 bn^2/p = O^*(b/p)$. (If there are more than $\rho_2 bn^2/p$ elements in our residue class, we know that r is not a good residue and abort the loop.)

- **Case 2:** $P = b^{2/3}a^{-1/3}$**.** In this case, we sample $O^*(b/\sqrt{ap})$ half-partitions drawn without replacement to create $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$.

The final step of Case 1 is to search $\boldsymbol{L}_1 \times \boldsymbol{L}_2$ for a matching pair. The Meet-in-the-Middle procedure takes time $O^*(\max(|\boldsymbol{L}_1|, |\boldsymbol{L}_2|, \text{ps}(\boldsymbol{L}_1, \boldsymbol{L}_2))$, where $\text{ps}(\boldsymbol{L}_1, \boldsymbol{L}_2)$ counts the number of *pseudosolutions*: pairs of half-partitions $(\mathbb{T}_1, \mathbb{T}_2) \in \boldsymbol{L}_1 \times \boldsymbol{L}_2$ such that $\text{sig}(\mathbb{T}_1) + \text{sig}(\mathbb{T}_2) = \tau - z\sum_i x_i$ but $(\mathbb{T}_1, \mathbb{T}_2)$ is not necessarily a matching pair (due to overlapping elements). Each pseudosolution must be checked and rejected. Using linearity of expectation over all distinct pairs of half-partitions in $B(\sigma)$, the expected number of pseudosolutions over a random input $\vec{\boldsymbol{x}}$ and choice of p, r is

$$O\left(\frac{b^2}{Mp}\right) = \frac{b}{p} \cdot e^{-\Omega(n)},$$

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

where it follows from (4.9) that b is smaller than $M \in |C|^{(1\pm\epsilon)n}$ by $e^{\Omega(n)}$. Hence by Markov's inequality, the probability that processing pseudosolutions takes time $\Omega(b/p)$, longer than the time to sample the residue class, is exponentially small. (If this occurs, the algorithm aborts and returns "no solution".) In Case 2, the expected number of pseudosolutions after subsampling is

$$O\left(\frac{b^2}{Mp}\cdot\left(\frac{b/\sqrt{ap}}{b/p}\right)^2\right) \le O\left(\frac{b}{p}\cdot\frac{p}{a}\right)\cdot e^{-\Omega(n)} \le O\left(\frac{b}{a}\right)\cdot e^{-\Omega(n)}$$

so processing pseudosolutions takes time $O(b/a)$ with very high probability.

The total runtime is thus the time it takes to create the table plus the time to sample and search for a matching pair. In Case 1, our runtime is:

$$O^*(\max(p, b/p)) = O^*(\max(a, b/a)) = O^*(b/a), \tag{4.11}$$

where the first equality follows from the fact that $p = \Theta(a/n)$ in Case 1 and the second equality follows from the fact that $a/n \le b^{2/3}a^{-1/3}$ in Case 1. In Case 2, our runtime is

$$O^*(\max(p, b/\sqrt{ap}, b/a)) = O^*(b^{2/3}a^{-1/3}). \tag{4.12}$$

Thus the algorithm has running time at most $O^*(\max(b/a, b^{2/3}a^{-1/3})$.

Let $\alpha := |\sigma|/n$. Recall from (4.8) and (4.9) that $a(\sigma) = \Theta^*(2^{|\sigma|})$ and that $b(\sigma) = O^*(2^{hn})$, where

$$h = H\left(\frac{\alpha}{2(|C|-1)}, \dots, \frac{\alpha}{2(|C|-1)}, 1-\frac{\alpha}{2}\right) = \frac{\alpha}{2}\cdot\log_2\left(\frac{2(|C|-1)}{\alpha}\right) + \left(1-\frac{\alpha}{2}\right)\cdot\log_2\left(\frac{2}{2-\alpha}\right).$$

It follows from these two equations that $b^{2/3}a^{-1/3} \le a/n$ when $|C| \le 3$, and thus we are always in Case 2 when $|C| \le 3$.

Assuming (4.7) without loss of generality, we have that α satisfies $\alpha \le 1 - 1/|C| + O(\delta)$. Using (4.8) and (4.9), we have the following bounds on b/a and $b^{2/3}a^{-1/3}$:

$$\frac{b}{a} \le 2^{H_1(\alpha)n}, \quad \text{where } H_1(\alpha) = \frac{\alpha}{2}\cdot\log_2\left(\frac{2|C|-2}{\alpha}\right) + \frac{2-\alpha}{2}\cdot\log_2\left(\frac{2}{2-\alpha}\right) - \alpha$$

and

$$\frac{b^{2/3}}{a^{1/3}} \le 2^{H_2(\alpha)n}, \quad \text{where } H_2(\alpha) = \frac{\alpha}{3}\cdot\log_2\left(\frac{2|C|-2}{\alpha}\right) + \frac{2-\alpha}{3}\cdot\log_2\left(\frac{2}{2-\alpha}\right) - \frac{\alpha}{3}.$$

When $|C| \ge 4$, in the range $(0, 1-1/|C|+O(\delta)]$ the maximum of H_1 and H_2 (achieved by H_1) is

$$H_1\big(1-1/|C|+O(\delta)\big) \le \log_2|C| + \frac{1}{C} - \frac{|C|+1}{2|C|}\log_2(|C|+1) + O(\delta^2).$$

When $|C| \le 3$, in the range $(0, 1-1/|C|+O(\delta)]$, the maximum (achieved by H_2) is

$$H_2\big(1-1/|C|+O(\delta)\big) \le \frac{2}{3}\log_2|C| - \frac{|C|+1}{3|C|}\log_2\left(\frac{|C|+1}{2}\right) + O(\delta^2).$$

Therefore, the runtime of the algorithm is $|C|^{\Lambda(n)n+O(\delta n)}$. □

5 Generalizations and Related Problems

5.1 Other Coefficient Sets Theorem 1 applies to $C = C(d)$ and $C = C_0(d)$, but our results can be applied to GSS on scale multiples and translations of $C(d)$ and $C_0(d)$. To see this, observe that for any integers α, β, an instance of GSS on the coefficient set $\alpha C + \beta$ is equivalent to GSS on C with target $\alpha^{-1}(t - \beta\sum_i x_i)$.

5.2 Number Balancing The Number Balancing problem attempts to divide n real numbers in $[0, 1]$ into two sets in a way that minimizes the difference between the two sums. The problem can be thought of as the optimization version of GSS on $C = \{0, \pm 1\}$. We introduce the following generalized version.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Problem 3: Generalized Number Balancing (GNB)

Input. A vector $\vec{y} = (y_1, y_2, \ldots, y_n) \in [0,1]^n$, a coefficient set $C \subset \mathbb{Z}$, and a precision $\delta > 0$.
Output. A coefficient vector $\vec{c} \in C^n$ that satisfies $|\vec{c} \cdot \vec{y}| \leq \delta$, or "no solution" if no solution exists.

In the average-case version of this problem, we consider inputs sampled uniformly at random from $[0,1]$. In [KK82], Karmarkar and Karp demonstrate a linear-time algorithm for worst-case Number Balancing that achieves precision $\delta = n^{-\Omega(\log(n))}$. However, a solution with exponentially small precision always exists by the pigeonhole principle. Scaling an average-case GNB instance $\vec{y}$ by δ^{-1} and truncating the result yields a vector of integers that can be interpreted as an instance $\vec{x}$ of GSS sampled uniformly from $[0 : \delta^{-1} - 1]^n$. A solution to $\vec{x}$ on C with target $\tau = 0$ is then a solution to $\vec{y}$ on C with precision δn. This insight yields the following corollary to Theorem 4 (as well as analogues for Theorem 3 and Corollary 2, omitted here for brevity.)

Corollary 3 (Optimal Precision for GNB with $C = C_0(d)$)**.** *Fix $C = C_0(d)$ and any $\epsilon > 0$, and consider $\vec{y} \sim [0,1]^n$. Then we have*

$$\Pr_{\vec{y}}\left[\exists \vec{c} \in C^n : |\vec{y} \cdot \vec{c}| < \delta n\right] \begin{cases} = 1 - e^{-\Omega(n)} & \text{if } \delta = \Omega^*(|C|^{-(1-\epsilon)n}) \\ \leq \delta |C|^n & \text{if } \delta \leq |C|^{-n}. \end{cases}$$

Moreover, the reduction from GNB to GSS allows us to use our algorithm to solve average-case GNB on symmetric coefficient sets.

Corollary 4 (Algorithm for Average-Case GNB)**.** *For any $\alpha \in (0,1)$, $C = C(d)$ or $C = C_0(d)$, and any constant $\epsilon > 0$, there exists an algorithm that solves average-case GNB with precision $|C|^{-\alpha n}$ in time*

$$O(|C|^{\alpha \Lambda(|C|)n + \epsilon n}),$$

where $\Lambda(n)$ is defined as in Theorem 1. For uniform random $\vec{y} \in [0,1]^n$, the algorithm is correct with probability at least $1 - e^{-\Omega(n)}$ for $C = C_0(d)$ and $1 - o_n(1)$ for $C = C(d)$.

Proof. We can convert $\vec{y} \sim [0,1]^n$ into $\vec{x} \sim [0 : n|C|^{\alpha n} - 1]^n$ by scaling and then truncating the input. (Note that this preserves uniform sampling.) Our structural results then guarantee the existence of a solution to this GSS instance with probability $1 - e^{-\Omega(n)}$ or $1 - o_n(1)$ depending on whether $C = C_0(d)$ or $C = C(d)$.[10] If a solution exists, it corresponds to a GNB solution with precision $|C|^{-\alpha n}$ and can be recovered by Algorithm 1 with probability $1 - e^{-\Omega(n)}$ in time $O(|C|^{\alpha \Lambda(|C|)n + \epsilon n})$ by Theorem 2. □

References

[AKKM13] Per Austrin, Petteri Kaski, Mikko Koivisto, and Jussi Määttä. Space–time tradeoffs for subset sum: An improved worst case algorithm. In *International Colloquium on Automata, Languages, and Programming*, pages 45–56. Springer, 2013.

[AKKN15] Per Austrin, Petteri Kaski, Mikko Koivisto, and Jesper Nederlof. Subset sum in the absence of concentration. In *32nd International Symposium on Theoretical Aspects of Computer Science (STACS 2015)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2015.

[AKKN16] Per Austrin, Mikko Koivisto, Petteri Kaski, and Jesper Nederlof. Dense subset sum may be the hardest. *33rd Symposium on Theoretical Aspects of Computer Science (STACS 2016)*, pages 13:1–13:14, 2016.

[BBSS20] Xavier Bonnetain, Rémi Bricout, André Schrottenloher, and Yixin Shen. Improved classical and quantum algorithms for subset-sum. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 633–666. Springer, 2020.

[BCJ11] Anja Becker, Jean-Sébastien Coron, and Antoine Joux. Improved generic algorithms for hard knapsacks. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 364–385. Springer, 2011.

[BCP01] Christian Borgs, Jennifer Chayes, and Boris Pittel. Phase transition and finite-size scaling for the integer partitioning problem. *Random Structures & Algorithms*, 19(3-4):247–288, 2001.

[10] If $C = \{\pm 1\}$ and $|\vec{x}|_1$ has odd parity, a solution that achieves target $\tau = 1$ is fine.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

[BGNV18] Nikhil Bansal, Shashwat Garg, Jesper Nederlof, and Nikhil Vyas. Faster space-efficient algorithms for subset sum, k-sum, and related problems. *SIAM J. Comput.*, 47(5):1755–1777, 2018.

[Böh11] E. Böhme. Verbesserte subset-sum algorithmen, 2011.

[DDKS12] Itai Dinur, Orr Dunkelman, Nathan Keller, and Adi Shamir. Efficient dissection of composite problems, with applications to cryptanalysis, knapsacks, and combinatorial search problems. In *Annual Cryptology Conference*, pages 719–740. Springer, 2012.

[HGJ10] Nick Howgrave-Graham and Antoine Joux. New generic algorithms for hard knapsacks. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 235–256. Springer, 2010.

[HS74] Ellis Horowitz and Sartaj Sahni. Computing partitions with applications to the knapsack problem. *Journal of the ACM (JACM)*, 21(2):277–292, 1974.

[KK82] Narendra Karmarkar and Richard M Karp. *The differencing method of set partitioning.* Computer Science Division (EECS), University of California Berkeley, 1982.

[Lue98] George S Lueker. Exponentially small bounds on the expected optimum of the partition and subset sum problems. *Random Structures & Algorithms*, 12(1):51–62, 1998.

[MNPW19] Marcin Mucha, Jesper Nederlof, Jakub Pawlewicz, and Karol Wegrzycki. Equal-subset-sum faster than the meet-in-the-middle. In Michael A. Bender, Ola Svensson, and Grzegorz Herman, editors, *27th Annual European Symposium on Algorithms, ESA 2019, September 9-11, 2019, Munich/Garching, Germany*, volume 144 of *LIPIcs*, pages 73:1–73:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

[Woe08] Gerhard J Woeginger. Open problems around exact algorithms. *Discrete Applied Mathematics*, 156(3):397–405, 2008.

A Proof of Corollary 2

Let $C = C(1)$. We first consider the upper bound when $M \geq |C|^n = 2^n$. Given any fixed $\vec{c} \in C^n$, conditioned on any values for $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{n-1}$, the probability that $\boldsymbol{x}_n$ is such that $\vec{\boldsymbol{x}} \cdot \vec{c} \in \{\tau, \tau+1\}$ is at most $2/M$. Union-bounding over all coefficient vectors gives the result.

We proceed to prove the lower bound for $M \leq |C|^{(1-\epsilon)n} = 2^{(1-\epsilon)n}$ for some $\epsilon > 0$. We start with the case in which $|\tau| \leq M$ and then extend our analysis to all τ such that $|\tau| = o(Mn)$.

Fix a target τ such that $|\tau| \leq M$. Let $\mathcal{Z}_{n,\tau}$ denote the expected number of solutions $\vec{c} \in C^n$ of $\vec{c} \cdot \vec{\boldsymbol{x}} = \tau$ over $\vec{\boldsymbol{x}} \sim [0 : M-1]^n$. In this case, [BCP01] Proposition 3.1 together with $M \leq 2^{(1-\epsilon)n}$ implies the following bounds on $\mathcal{Z}_{n,\tau}$:

$$\mathbf{E}\left[\mathcal{Z}_{n,\tau}\right] = \rho_n(1 + O(n^{-1})) \quad \text{and} \quad \mathbf{E}\left[\mathcal{Z}_{n,\tau}^2\right] = 2\rho_n^2(1 + O(n^{-1})).$$

where ρ_n is defined as

$$\rho_n := \sqrt{\frac{3}{2\pi n}} \cdot \frac{2^n}{M}.$$

Note that we cannot directly apply Chebyshev's inequality to obtain concentration of $\mathcal{Z}_{n,\tau}$ because of the extra factor of 2 in $\mathbf{E}[\mathcal{Z}_{n,\tau}]^2$. (The factor of 2 is there because of the observation that $\mathcal{Z}_{n,\tau}$ can have a solution only when the sum of $\vec{\boldsymbol{x}}$ is even, which happens with probability $1/2$ over $\vec{\boldsymbol{x}}$.)

Since we are interested in the probability of having $\vec{c} \in C^n$ with $\vec{c} \cdot \vec{x} \in \{\tau, \tau+1\}$, we have

$$\mathbf{E}\left[\mathcal{Z}_{n,\tau} + \mathcal{Z}_{n,\tau+1}\right] = 2\rho_n(1 + O(n^{-1})) \quad \text{and} \quad \mathbf{E}\left[(\mathcal{Z}_{n,\tau} + \mathcal{Z}_{n,\tau+1})^2\right] = 4\rho_n^2(1 + O(n^{-1})),$$

where the second equation used the fact that $\mathbf{E}[\mathcal{Z}_{n,\tau}\mathcal{Z}_{n,\tau+1}] = 0$, because for any $\vec{x}$ one can never have a solution for both τ and $\tau+1$ due to the parity issue. It follows from Chebyshev's inequality that $\mathcal{Z}_{n,\tau} + \mathcal{Z}_{n,\tau+1} > 0$ with probability $1 - o_n(1)$ (see the beginning of the proof of Theorem 3).

To extend the result to a larger range of offsets, we fix $|\tau| = o(Mn)$, and assume without loss of generality that τ is positive. Consider the experiment in which input elements $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots$ are revealed one by one and assigned the "-1" coefficient. We stop at $\boldsymbol{x}_i$ when $\boldsymbol{x}_1 + \ldots + \boldsymbol{x}_i \in [\tau, \tau+M]$ for the first time. Standard concentration arguments show that this process stops with less than $\epsilon n/2$ elements revealed with high probability (using $|\tau| = o(Mn)$). When this happens, we reduce the problem to a new GSS instance with target $|\tau'| \leq M$ and $n' \geq (1-\epsilon/2)n$ elements. The latter implies that $M \leq |C|^{(1-\epsilon/2)n'}$ so this reduces to our earlier analysis (with ϵ there replaced by $\epsilon/2$).

B Proof of Theorem 2

The proof of Theorem 2 is similar to the proof of Theorem 1 and follows a similar intuition: if $M \leq |C|^{(1-\epsilon)n}$ for any $\epsilon > 0$, then we can reduce the instance size to n' so that M falls inside the window $|C|^{(1\pm\epsilon)n'}$ and the

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

algorithm of Theorem 5 applies. Moreover, shrinking the instance size results in a faster running time.

Proof of Theorem 2. Fix $d \in \mathbb{N}_{\geq 1}$, $C = C(d)$ or $C = C_0(d)$, and τ such that $|\tau| = o(Mn)$. Consider $M = |C|^{\alpha n + o(n)}$ for some $\alpha \in (0,1)$. Fix a constant ζ as specified in the statement of Theorem 2. We proceed to show an algorithm for average-case GSS with running time $|C|^{\alpha\Lambda(|C|)n+\zeta n}$.

As $M = |C|^n \cdot 2^{-\Omega(n)}$, we can use the procedures described in the proof of Theorem 1 to create a new instance $\vec{\boldsymbol{x}}'$ of average-case GSS with n' elements, where n' satisfies

$$|C|^{(1-\epsilon_1)n'} \leq M \leq |C|^{(1-\epsilon_1/2)n'} \tag{B.1}$$

for a constant $\epsilon_1 > 0$ that can be made arbitrarily small.

In the $C = C_0(d)$ case, we can simply set $\vec{\boldsymbol{x}}' = \vec{\boldsymbol{x}}_{[n']}$. In the $C = C(d)$ case, we perform the shrinking operation described in the proof of Theorem 1, assigning $+1$ and -1 coefficients to the elements of $\vec{\boldsymbol{x}}_{[n'+1:n]}$ and adjusting the target τ accordingly. In both cases, we create a new instance $\vec{\boldsymbol{x}}'$, uniformly distributed over $[0 : M-1]^{n'}$, with a new target τ' such that $|\tau'| = o(n'M)$, and for which any solution can be easily converted to a solution for $\vec{\boldsymbol{x}}$.

Set $\xi \leq \zeta/2$ to be sufficiently small that Theorem 5 holds on $\vec{\boldsymbol{x}}'$, and let $\epsilon > 0$ be the constant determined by ξ in Theorem 5. Run the algorithm in Theorem 5 on $\vec{\boldsymbol{x}}'$ for every profile π and return any solution found (or "no solution" if no solution is found for any profile π.) Because there are polynomially many solution profiles, this takes time

$$O^*(|C|^{\Lambda(|C|)n'+\xi n}) = O^*(|C|^{(\alpha\Lambda(|C|)/(1-\epsilon_1)+\zeta/2)n+o(n)}),$$

where we use the fact that $n' \leq \alpha/(1-\epsilon_1)$, which follows from Equation (B.1). For sufficiently small ϵ_1, this is dominated by $|C|^{\alpha\Lambda(|C|)n+\zeta n}$.

Taking a union bound over the chance that the algorithm in Theorem 5 fails on any solution profile yields a success probability of $1 - e^{-\Omega(n)}$ on $\vec{\boldsymbol{x}}'$. In the $C = C_0(d)$ case, $\vec{\boldsymbol{x}}'$ has a solution with probability $1 - e^{-\Omega(n)}$ by Theorem 4, and thus we solve $\vec{\boldsymbol{x}}$ with probability $1 - e^{-\Omega(n)}$. By Theorem 3, there is a solution with probability at least $1 - o_n(1)$ over the randomness of $\vec{\boldsymbol{x}}'$ in the $C = C(d)$, $d > 1$ case. By Corollary 2, there is a solution with probability at least $1 - o_n(1)$ over the randomness of $\vec{\boldsymbol{x}}'$ in the $C = C(1)$ case if $\sum_i \boldsymbol{x}_i$ has the same parity as τ, as the shrinking procedure preserves the parity of $\sum_i \boldsymbol{x}_i - \tau$. Thus we solve $\vec{\boldsymbol{x}}$ with probability $1 - o_n(1)$ in this case. □

C Proof of Theorem 3

Here the challenge is again to prove the lower bound; the proof of the upper bound on solution probability is trivial and follows the same argument as that in the proof of Theorem 4.

To prove the lower bound, consider a coefficient set $C = C(d)$ for a fixed constant $d > 1$, $M = O^*(|C|^{(1-\epsilon)n})$ for a fixed constant $\epsilon > 0$, and a target τ satisfying $|\tau| = O(M)$. (We expand our discussion to all τ such that $|\tau| = o(Mn)$ at the end of the subsection.) Define

$$\rho_n := \frac{|C|^{n+1/2}}{M\sqrt{2\pi n\kappa \sum_{c\in C} c^2}} \qquad \text{where} \qquad \kappa := \mathop{\mathbf{E}}_{\boldsymbol{\xi}\sim[0:M-1]}\left[\frac{\boldsymbol{\xi}^2}{M^2}\right] = \frac{1}{3} - \frac{1}{2M} + \frac{1}{6M^2}. \tag{C.2}$$

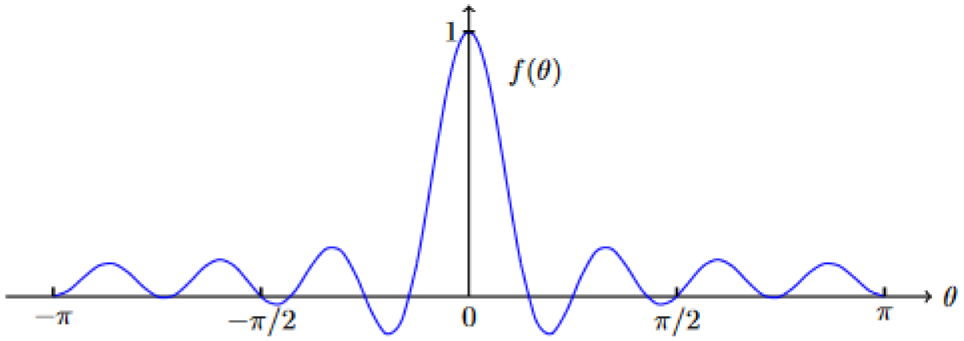
Because $|C| = 2d$ and $\sum_{c\in C} c^2 = \frac{2}{3}d(d+1)(2d+1)$ are fixed constants, we have $\rho_n = \Theta(\frac{|C|^n}{M\sqrt{n}})$. Moreover, because $M = O^*(|C|^{(1-\epsilon)n})$ for a constant $\epsilon > 0$, we have $\rho_n = |C|^{\Omega(n)}$.

Let $\boldsymbol{\mathcal{Z}} := \boldsymbol{\mathcal{Z}}(M, C, \tau)$ be the random variable that counts the number of solution vectors $\vec{c} \in C^n$ for a random ATSS instance $\vec{\boldsymbol{x}} \sim [0 : M-1]^n$ with target value τ. Our proof of the lower bound on solution probability when $M = O^*(|C|^{(1-\epsilon)n})$ generalizes [BCP01, Proposition 3.1], and consists of three parts:

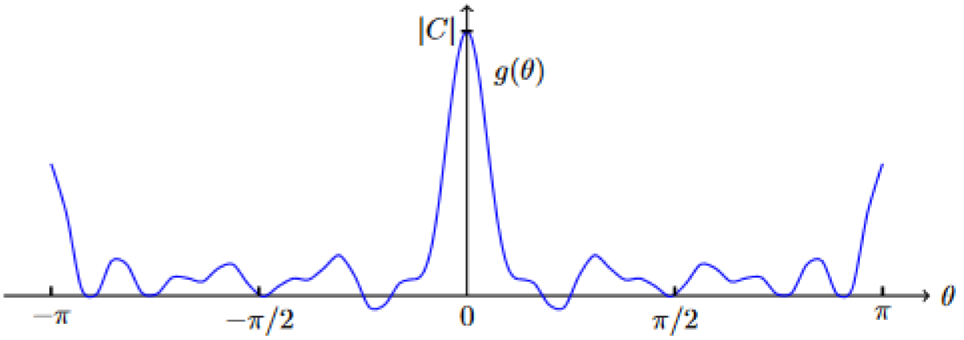
1. Lemma 5 proves that $\mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}] = \rho_n \cdot (1 \pm o_n(1))$.
2. Lemma 6 proves that $\mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}^2] \leq \rho_n^2 \cdot (1 + o_n(1))$.
3. In the proof of Theorem 3, we use the preceding lemmas to show $\boldsymbol{\mathcal{Z}} = \rho_n \cdot (1 \pm o_n(1))$ with probability $1 - o_n(1)$. The bound for $M = O^*(|C|^{(1-\epsilon)n})$ follows.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy



(a) f with $M = 8$, $C = C(2)$.



(b) g with $M = 8$, $C = C(2)$.

Figure 2: Example plots of f and g. As $|C|$ increases, so does the complexity of the oscillation.

The first two steps are accomplished by expressing $\boldsymbol{\mathcal{Z}}$ as an integral written in terms of a certain function $g : [-\pi, \pi] \to \mathbb{R}$ to the nth power. In both cases, we show that the mass of g^n is highly concentrated in a region near the origin and tightly bound the value of the function in this region. In particular, we will employ the function $f : [-\pi, \pi] \to \mathbb{R}$,

$$f(\theta) := \mathop{\mathbf{E}}_{\boldsymbol{\xi} \sim [0:M-1]} [\cos(\theta \boldsymbol{\xi})] = \frac{1}{M} \sum_{j \in [0:M-1]} \cos(j\theta) = \frac{1}{M} \Big(\frac{\sin((M - 1/2)\theta)}{2 \sin(\theta/2)} + \frac{1}{2} \Big), \tag{C.3}$$

which is defined almost identically[11] to the function f that occurs in the proof of [BCP01, Proposition 3.1]. The last equality corresponds to [BCP01, Equation (3.11)].

Proof of Theorem 3 (Assuming Lemmas 5 and 6). We have

$$\begin{aligned}
\mathbf{E}_{\vec{\boldsymbol{x}}} \Big[\big| \boldsymbol{\mathcal{Z}}/\rho_n - 1 \big| \Big] &\le \sqrt{\mathbf{E}_{\vec{\boldsymbol{x}}} \Big[\big(\boldsymbol{\mathcal{Z}}/\rho_n - 1 \big)^2 \Big]} \\
&= \frac{1}{\rho_n} \sqrt{\mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}^2] - \mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}]^2 + \Big(\mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}] - \rho_n \Big)^2} \\
&\le \frac{1}{\rho_n} \sqrt{\mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}^2] - \mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}]^2} + \frac{1}{\rho_n} \Big| \mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}] - \rho_n \Big|
\end{aligned}$$

[11] The difference occurs because [BCP01] consider inputs drawn uniformly from the set $[M]$, while we consider inputs drawn uniformly from the set $[0 : M - 1]$.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

$$= o_n(1).$$

Here the first step holds since any random variable $\boldsymbol{\mathcal{X}}$ satisfies that $\mathbf{E}[\boldsymbol{\mathcal{X}}]^2 \le \mathbf{E}[\boldsymbol{\mathcal{X}}^2]$, the second step is elementary algebra, the third step holds since $\sqrt{a+b} \le \sqrt{a} + \sqrt{b}$ for any $a, b \in \mathbb{R}_{\ge 0}$, and the last step uses Lemma 5 that $\mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}] = \rho_n \cdot (1 \pm o_n(1))$ and Lemma 6 that $\mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}^2] \le \rho_n^2 \cdot (1 \pm o_n(1))$.

Our instance has a solution unless $|\boldsymbol{\mathcal{Z}}/\rho_n - 1| > 1$, which occurs with probability at most $o_n(1)$ by Markov's inequality. Accordingly, for $C = C(d)$, $d > 1$, $|\tau| = O(M)$, and $M = O^*(|C|^{(1-\epsilon)n})$ for any constant $\epsilon > 0$, we have

$$\mathbf{Pr}_{\vec{\boldsymbol{x}}}\left[\exists \vec{c} \in C^n : \vec{\boldsymbol{x}} \cdot \vec{c} = \tau\right] = 1 - o_n(1).$$

It remains to consider the full range of targets τ. Consider a randomly sampled GSS instance as above but with $|\tau| = o(Mn)$. Without loss of generality, consider $\tau > 0$ and consider the experiment in which we sample input elements one by one and assign each the coefficient -1. Each step of this process effectively creates a new instance with one fewer input element and a smaller target τ'. It is a simple exercise to show that when $\tau = o(Mn)$, our new instance satisfies $\tau' = O(M)$ with high probability after $o(n)$ steps. This establishes the lower bound on the probability that a solution exists in Theorem 3.

Finally, we consider the upper bound on solution probability. For any coefficient vector $\vec{c} \in C^n$, we observe that it solves at most a $1/M$ fraction of GSS instances. This follows from the fact that for any fixed set of $n-1$ input elements in $[0 : M-1]^{n-1}$, there is at most one choice for the last element such that $\vec{c}$ solves the instance. Union-bounding over all coefficient vectors yields that the number of instances that admit any solution is at most $|C|^n \cdot M^{n-1}$. Dividing by the size of the instance space yields

$$\mathbf{Pr}_{\vec{\boldsymbol{x}}}\left[\exists \vec{c} \in C^n : \vec{\boldsymbol{x}} \cdot \vec{c} = \tau\right] \le \frac{|C|^n}{M}. \qquad \square$$

C.1 Expectation of $\boldsymbol{\mathcal{Z}}$

Lemma 5 (Expectation of $\boldsymbol{\mathcal{Z}}$). *Fix a coefficient set $C = C(d) = \{\pm 1, \pm 2, \dots, \pm d\}$ for some integer $d > 1$, $M = O^*(|C|^{(1-\epsilon)n})$ for some constant $\epsilon > 0$, and a target $\tau = O(M)$. For ρ_n and f defined as in Equation (C.2) and Equation (C.3), we have*

$$\mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \cos(\tau\theta) \cdot g(\theta)^n \cdot \mathrm{d}\theta = \rho_n \cdot (1 \pm o_n(1)),$$

where the function $g(\theta) := \sum_{c \in C} f(c\theta)$.

For any choice of $\vec{\boldsymbol{x}} \sim [0 : M-1]^n$, we observe that

$$\begin{aligned}
\boldsymbol{\mathcal{Z}} &= \sum_{\vec{c} \in C^n} \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{\mathrm{i}\theta(\vec{\boldsymbol{x}} \cdot \vec{c} - \tau)} \cdot \mathrm{d}\theta \\
&= \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{-\mathrm{i}\theta\tau} \sum_{\vec{c} \in C^n} \Big(\prod_{j \in [n]} e^{\mathrm{i}\theta c_j \boldsymbol{x}_j} \Big) \cdot \mathrm{d}\theta \\
&= \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{-\mathrm{i}\theta\tau} \prod_{j \in [n]} \Big(\sum_{c \in C} e^{\mathrm{i}\theta c \boldsymbol{x}_j} \Big) \cdot \mathrm{d}\theta,
\end{aligned}$$

where the first step chooses an integral to act as an indicator for the event $\{\vec{\boldsymbol{x}} \cdot \vec{c} = \tau\}$, the second step factors the dot product,[12] and the last step exchanges the sum and the product (which is enabled by the special form of the integrand).

We note that $\sum_{c \in C} e^{\mathrm{i}c\theta \boldsymbol{x}_j} = \sum_{c \in [d]} (e^{\mathrm{i}c\theta \boldsymbol{x}_j} + e^{-\mathrm{i}c\theta \boldsymbol{x}_j}) = \sum_{c \in [d]} 2\cos(c\theta \boldsymbol{x}_j) = \sum_{c \in C} \cos(c\theta \boldsymbol{x}_j)$, by the fact that $C = C(d) = \{\pm 1, \pm 2, \dots, \pm d\}$, the Euler formula, and cancelling sines. This identity allows us to simplify our equation for $\boldsymbol{\mathcal{Z}}$ to

$$\boldsymbol{\mathcal{Z}} = \frac{1}{2\pi} \int_{-\pi}^{\pi} \cos(\tau\theta) \prod_{j \in [n]} \Big(\sum_{c \in C} \cos(c\theta \boldsymbol{x}_j) \Big) \cdot \mathrm{d}\theta. \tag{C.4}$$

[12] We have assumed $C = C(d) = \{\pm 1, \pm 2, \dots, \pm d\}$, so we need not worry about excluding the $\vec{\mu}(C)$ solution. However, we can generalize this to the case where $\vec{\mu}(C) \in C^n$ by subtracting one from this equation.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Since all the $\boldsymbol{x}_j \sim [0 : M-1]$ are i.i.d., we can break down the expectation $\mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}]$ as follows:

$$\begin{aligned}\mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}] &= \frac{1}{2\pi}\int_{-\pi}^{\pi} \cos(\tau\theta)\cdot\Big(\mathop{\mathbf{E}}_{\boldsymbol{\xi}\sim[0:M-1]}\Big[\sum_{c\in C}\cos(c\theta\boldsymbol{\xi})\Big]\Big)^n \mathrm{d}\theta \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} \cos(\tau\theta)\cdot g(\theta)^n\cdot \mathrm{d}\theta,\end{aligned}$$

where the last step follows because $f(c\theta) = \mathbf{E}_{\boldsymbol{\xi}\sim[0:M-1]}[\cos(c\theta\boldsymbol{\xi})]$ (see (C.3)) and $g(\theta) = \sum_{c\in C} f(c\theta)$. Later we will see that the mass of this integral is highly concentrated around the origin, where $\cos(\tau\theta)\cdot g(\theta)^n \approx |C|^n$.

Fix any a and b satisfying $|C|^2/\epsilon < a < b$. In the rest of this proof, let us consider sufficiently large $n \geq 6$ and $M \geq 20$ such that

$$\sqrt{64\ln(n)/n} \leq 1 \leq |C| \qquad \text{and} \qquad M \geq \max\Big\{8d\cdot b,\ \frac{a\cdot b}{2b-2a}\Big\}. \tag{C.5}$$

Then the quantity b_0 below is well defined (as $0 \leq b/(2M) \leq 1/d \leq 1$) and satisfies $b \leq b_0 \leq (\pi/2)\cdot b$.

$$b_0 := 2M\cdot \sin^{-1}\Big(\frac{b}{2M}\Big). \tag{C.6}$$

To evaluate the expectation $\mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}]$, we readopt the approach by [BCP01] and split its formula into the following three parts. (Based on (C.5) and (C.6), we notice that $\sqrt{64\ln(n)/n} \leq |C| < b \leq b_0$ and $\frac{b_0}{M} \leq \frac{(\pi/2)\cdot b}{8d\cdot b} \leq \frac{\pi}{16}$. Hence, for each of these parts, the interval of integration is well defined.)

$$\text{(Part I)} \qquad \mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}] = \frac{1}{2\pi}\int_{|\theta|\leq \frac{\sqrt{64\ln(n)/n}}{M}} \cos(\tau\theta)\cdot g(\theta)^n\cdot \mathrm{d}\theta$$

$$\text{(Part II)} \qquad + \frac{1}{2\pi}\int_{|\theta|\in[\frac{\sqrt{64\ln(n)/n}}{M},\ \frac{b_0}{M}]} \cos(\tau\theta)\cdot g(\theta)^n\cdot \mathrm{d}\theta$$

$$\text{(Part III)} \qquad + \frac{1}{2\pi}\int_{|\theta|\in[\frac{b_0}{M},\ \pi]} \cos(\tau\theta)\cdot g(\theta)^n\cdot \mathrm{d}\theta.$$

Below let us quantify (Part I), (Part II), and (Part III) one by one.

Claim 3. $|(\text{Part I})| = \rho_n(1\pm o(1))$.

Proof. Following [BCP01], we set $y := M\theta$ and consider $|y| \leq \sqrt{64\ln(n)/n} = o_n(1)$ small enough. In this range, we observe that

$$\begin{aligned} f(\theta) = f(y/M) &= \mathop{\mathbf{E}}_{\boldsymbol{\xi}\sim[0:M-1]}\big[\cos(\boldsymbol{\xi}(y/M))\big] \\ &= \mathop{\mathbf{E}}_{\boldsymbol{\xi}\sim[0:M-1]}\Big[1-\frac{\boldsymbol{\xi}^2}{2}(y/M)^2 \pm O(y^4)\Big] \\ &= 1-\frac{\kappa}{2}y^2 \pm O(y^4) \\ &= \Big(1-\frac{\kappa}{2}y^2\Big)\cdot(1\pm O(y^4)), \end{aligned} \tag{C.7}$$

where the second step uses the Taylor series $\cos(z) = 1-\frac{1}{2}z^2 \pm O(z^4)$, the third step follows from the definition $\kappa = \mathbf{E}_{\boldsymbol{\xi}\sim[0:M-1]}[\frac{\boldsymbol{\xi}^2}{M^2}]$, and the last step converts the (additive) error term into a multiplicative form. We can write $g(y/M) = \sum_{c\in C} f(cy/M)$ in a similar form:

$$\begin{aligned} g(y/M) &= \Big(|C| - \frac{\kappa}{2}\Big(\sum_{c\in C}c^2\Big)y^2\Big)\cdot(1\pm O(y^4)) \\ &= |C|\cdot\exp\Big(-\frac{\kappa}{2|C|}\Big(\sum_{c\in C}c^2\Big)y^2\Big)\cdot(1\pm O(y^4)), \end{aligned}$$

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

where the last step uses the Taylor series $e^{-z} = 1 - z \pm O(z^2)$. It follows that

$$g(y/M)^n = |C|^n \cdot \exp\Big(- \frac{n\kappa}{2|C|}\Big(\sum_{c\in C} c^2\Big)y^2\Big) \cdot (1 \pm O(ny^4)). \tag{C.8}$$

Based on (C.8), we can evaluate (Part I) as follows:

$$\begin{aligned}
\text{(Part I)} &= \frac{1}{2\pi M}\int_{|y|\le\sqrt{64\ln(n)/n}} \cos(ty/M)\cdot g(y/M)^n \cdot \mathrm{d}y \\
&= \frac{1}{2\pi M}\int_{|y|\le\sqrt{64\ln(n)/n}} (1\pm O(y^2))\cdot g(y/M)^n \cdot \mathrm{d}y \\
&= \frac{|C|^n}{2\pi M}\int_{|y|\le\sqrt{64\ln(n)/n}} (1\pm O(y^2+ny^4))\cdot \exp\Big(- \frac{n\kappa}{2|C|}\Big(\sum_{c\in C} c^2\Big)y^2\Big)\mathrm{d}y \\
&= \frac{|C|^n}{2\pi M}\cdot(1\pm o_n(1))\cdot\int_{|y|\le\sqrt{64\ln(n)/n}} \exp\Big(- \frac{n\kappa}{2|C|}\Big(\sum_{c\in C} c^2\Big)y^2\Big)\mathrm{d}y \\
&= \frac{|C|^n}{2\pi M}\cdot(1\pm o_n(1))\cdot\sqrt{\frac{2\pi|C|}{n\kappa\sum_{c\in C}c^2}}\cdot \mathrm{erf}\left(\sqrt{\frac{32\kappa}{|C|}\Big(\sum_{c\in C}c^2\Big)\ln(n)}\right) \\
&= \rho_n\cdot(1\pm o_n(1))\cdot \mathrm{erf}\left(\sqrt{\frac{32\kappa}{|C|}\Big(\sum_{c\in C}c^2\Big)\ln(n)}\right).
\end{aligned}$$

Here the first step changes variable $y = M\theta$. The second step follows as $\cos(z) = 1 - O(z^2)$ and $|t| = O(M)$. The third step substitutes (C.8). The fourth step follows as $O(y^2 + ny^4) = o_n(1)$ when $|y| \le \sqrt{64\ln(n)/n}$. The fifth step resolves the integral by using the Gaussian error function $\mathrm{erf}(z)$. And the last step uses the definition of ρ_n (see (C.2)).

To finish the proof of Claim 3, it remains to reason about the Gaussian error function $\mathrm{erf}(z)$. It is known that $|1 - \mathrm{erf}(z)| \le e^{-z^2}$ for any $z \in \mathbb{R}_{\ge 0}$. As a consequence, we have

$$\mathrm{erf}\left(\sqrt{\frac{32\kappa}{|C|}\Big(\sum_{c\in C}c^2\Big)\ln(n)}\right) = 1 \pm n^{-32\kappa|C|^{-1}(\sum_{c\in C}c^2)} = 1 \pm n^{-32\kappa} = 1 \pm o_n(1), \tag{C.9}$$

where the second step holds because $C = C(d) = \{\pm1, \pm2, \ldots, \pm d\}$ and then $|C|^{-1}(\sum_{c\in C} c^2) \ge 1$, and the last step holds because $\kappa = \frac{1}{3} - \frac{1}{2M} + \frac{1}{6M^2} = \Omega_n(1)$.

Combining everything together completes the proof of Claim 3. □

Claim 4. $|\text{(Part II)}| = o(\rho_n)$.

Proof. Once again we set $y := M\theta$. Recall that (Part II) considers the range $|\theta| \in [\frac{\sqrt{64\ln(n)/n}}{M}, \frac{b_0}{M}]$, namely $|y| \in [\sqrt{64\ln(n)/n}, b_0]$. In this range, it turns out that

$$|f(c\theta)| = |f(cy/M)| \le \frac{1}{\sqrt[n]{n}}, \tag{C.10}$$

for any (nonzero) coefficient $c \in C$. Assuming the truth of (C.10) for the moment, we have

$$|g(\theta)| = |g(y/M)| \le \sum_{c\in C}|f(cy/M)| \le \frac{|C|}{\sqrt[n]{n}}, \tag{C.11}$$

for any $|y| \in [\sqrt{64\ln(n)/n}, b_0]$. Hence, the magnitude of (Part II) is at most

$$|\text{(Part II)}| = \left|\frac{1}{2\pi M}\int_{|y|\in[\sqrt{64\ln(n)/n},\, b_0]} \cos(ty/M)\cdot g(y/M)^n\cdot \mathrm{d}y\right|$$

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

$$
\begin{aligned}
&\le \frac{1}{2\pi M} \cdot 2b_0 \cdot \Big(\max_{|y| \in [\sqrt{64\ln(n)/n},\, b_0]} |g(y/M)| \Big)^n \\
&\le \frac{1}{2\pi M} \cdot 2b_0 \cdot \frac{|C|^n}{n} \\
&= \frac{b_0 |C|^n}{\pi M n} \\
&= o(\rho_n),
\end{aligned}
$$

where the first step changes variables using the equality $y = M\theta$, the third step applies (C.11), and the last step holds since $b_0 \le \frac{\pi}{2} \cdot b$ is a constant (see (C.6)) and $\rho_n = \Theta(\frac{|C|^n}{M\sqrt{n}})$ (see (C.2)).

It remains to verify (C.10). Before doing so, we observe that for any $|y| \le b_0$,

$$
\Big|\frac{cy}{2M}\Big| \le \frac{|c| \cdot b_0}{2M} \le \frac{|c| \cdot (\pi/2) \cdot b}{16d \cdot b} \le \frac{|c| \cdot (\pi/32)}{d} \le \frac{\pi}{32}, \tag{C.12}
$$

where the second step holds because $b_0 \le (\pi/2) \cdot b$ (see (C.6)) and $M \ge 8d \cdot b$ (see (C.5)), and the last step holds because $c \in C = C(d) = \{\pm 1, \pm 2, \ldots, \pm d\}$.

Thus for any $|y| \le b_0$, we can upper-bound $|f(c\theta)| = |f(cy/M)|$ as follows.

$$
\begin{aligned}
|f(c\theta)| = |f(cy/M)| &= \Big|\frac{\sin(cy - cy/(2M))}{2M\sin(cy/(2M))} + \frac{1}{2M}\Big| \\
&= \Big|\frac{\sin(cy)}{2M\tan(cy/(2M))} + \frac{1-\cos(cy)}{2M}\Big| \\
&\le \Big|\frac{\sin(cy)}{cy}\Big| + \frac{1-\cos(cy)}{2M} \\
&\le \Big|\frac{\sin(cy)}{cy}\Big| + \frac{1-\cos(cy)}{40}. \qquad \text{(C.13)}
\end{aligned}
$$

Here the first step changes variables using the equality $y = M\theta$. The second step applies the identity $\sin(\alpha - \beta) = \sin(\alpha)\cos(\beta) - \sin(\beta)\cos(\alpha)$. The third step holds since $|\tan(z)| \ge |z|$ for any $|z| \le \frac{\pi}{2}$ (notice that $|\frac{cy}{2M}| \le \frac{\pi}{4} \le \frac{\pi}{2}$, see (C.12)). And the last step follows since $M \ge 20$ (see (C.5)).

We prove (C.10) for any nonzero $c \in C$ and any $|y| \in [\sqrt{64\ln(n)/n},\, b_0]$ in two cases.

Case I: $\sqrt{64\ln(n)/n} \le |y| \le \frac{\pi}{2|c|}$**.** We know that $|\sin(z)/z| \le e^{-z^2/6}$ for any $|z| \le \frac{\pi}{2}$ and $\cos(z) \ge 1 - \frac{1}{2}z^2$ for any $z \in \mathbb{R}$. Applying both facts to (C.13) gives

$$
\begin{aligned}
|f(c\theta)| = |f(cy/M)| &\le \Big|\frac{\sin(cy)}{cy}\Big| + \frac{1-\cos(cy)}{40} \\
&\le e^{-\frac{1}{6}c^2y^2} + \frac{1}{80}c^2y^2 \\
&\le e^{-\frac{1}{8}c^2y^2} \\
&\le e^{-\frac{1}{8}y^2} \\
&\le \frac{1}{\sqrt[n]{n}},
\end{aligned}
$$

where the third step holds since $e^{-z^2/6} + \frac{1}{80}z^2 \le e^{-z^2/8}$ for any $|z| \le \frac{\pi}{2}$, the fourth step holds since $c \in C$ is a nonzero integer, and the last step holds when $|y| \ge \sqrt{64\ln(n)/n}$.

Case II: $\frac{\pi}{2|c|} \le |y| \le b_0$**.** Following (C.13), in this range we have

$$
|f(c\theta)| = |f(cy/M)| \le \Big|\frac{\sin(cy)}{cy}\Big| + \frac{1-\cos(cy)}{40}
$$

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

$$\le \frac{|\sin(cy)|}{\pi/2} + \frac{1+|\cos(cy)|}{40}$$
$$\le \frac{2}{\pi} + \frac{1}{20}$$
$$\le \frac{1}{\sqrt[n]{n}},$$

where the second step applies $|y| \ge \frac{\pi}{2|c|}$, the third step holds because $|\sin(cy)| \le 1$ and $|\cos(cy)| \le 1$, and the last step holds because $\frac{2}{\pi} + \frac{1}{20} \approx 0.6866$ and $\frac{1}{\sqrt[n]{n}} = 1 - o_n(1)$.

Combining both cases together gives (C.10). This completes the proof of Claim 4. □

Claim 5. $|(\text{Part III})| = o(\rho_n)$.

Proof. Recall that (Part III) considers the range $|\theta| \in [\frac{b_0}{M}, \pi]$, in which we have

$$|f(\theta)| = \Big|\frac{1}{M}\Big(\frac{\sin((M-1/2)\theta)}{2\sin(\theta/2)} + \frac{1}{2}\Big)\Big| \le \frac{1}{b} + \frac{1}{2M} \le \frac{1}{a}. \tag{C.14}$$

Here the first inequality holds as $|\sin((M-1/2)\theta)| \le 1$ and $|\sin(\theta/2)| \ge |\sin(\frac{b_0}{2M})| = \frac{b}{2M}$ (see (C.6)), and the second inequality holds as $M \ge \frac{ab}{2b-2a}$ (see (C.5)).

For any $c \in C$ and $|\theta| \in [\frac{b_0}{M}, \pi]$, we know from (C.3) that $|f(c\theta)| = |\mathbf{E}_{\boldsymbol{\xi}\sim[0:M-1]}[\cos(c\theta\boldsymbol{\xi})]| \le 1$. (Recall that $C = C(d) \supseteq \{\pm 1\}$.) Applying this fact and (C.14), for any $|\theta| \in [\frac{b_0}{M}, \pi]$ we have

$$|g(\theta)| \le |f(\theta)| + |f(-\theta)| + \sum_{c\in C\setminus\{\pm1\}} |f(c\theta)| \le \frac{2}{a} + (|C|-2) = |C|^{(1-\epsilon_1)} \tag{C.15}$$

for some constant $\epsilon_1 := \epsilon_1(|C|, a) \in (0,1)$, where the last step holds since $a > |C|^2/\epsilon > 1$ (see (C.5)). Further, we know that for any integer $c \in C$ the function $f_c(\theta) := f(c\theta)$ is 2π-periodic (see (C.3)). Thus (C.14) implies that

$$\Big|\Big\{\theta \in [-\pi,\pi] : |f(c\theta)| > \frac{1}{a}\Big\}\Big| \le \frac{2b_0}{M}. \tag{C.16}$$

This, given that $g(\theta) = \sum_{c\in C} f(c\theta)$, further gives

$$\Big|\Big\{\theta \in [-\pi,\pi] : |g(\theta)| > \frac{|C|}{a}\Big\}\Big| \le \frac{2b_0|C|}{M}. \tag{C.17}$$

By considering separately the subregion on which $|C|/a < |g(\theta)| \le |C|^{(1-\epsilon_1)}$ and the subregion on which $|g(\theta)| \le |C|/a$, the magnitude of (Part III) is at most

$$|(\text{Part III})| = \Big|\frac{1}{2\pi}\int_{|\theta|\in[\frac{b_0}{M},\,\pi]} \cos(\tau\theta)\cdot g(\theta)^n \cdot \mathrm{d}\theta\Big|$$
$$\le \frac{1}{2\pi}\int_{|\theta|\in[\frac{b_0}{M},\,\pi]} |g(\theta)|^n \cdot \mathrm{d}\theta$$
$$\le \frac{1}{2\pi}\Big(|C|^{(1-\epsilon_1)n}\cdot\frac{2b_0|C|}{M} + \Big(\frac{|C|}{a}\Big)^n\cdot 2\pi\Big)$$
$$= O\Big(\frac{|C|^{(1-\epsilon_1)n}}{M}\Big) + O\Big(\frac{1}{|C|^n}\Big),$$

where the third step applies (C.15) and (C.17), and the fourth step holds since (for the first term) both b_0 and $|C|$ are constants and (for the second term) $a > |C|^2/\epsilon > |C|^2$ (see (C.5)).

Recall (C.2) that $\rho_n = |C|^{\Omega(n)}$. So the above two terms are upper-bounded by $O(|C|^{(1-\epsilon_1)n}/M) = \rho_n \cdot O(\sqrt{n}\cdot|C|^{-\epsilon_1 n}) = o(\rho_n)$ and $O(|C|^{-n}) = o_n(1) = o(\rho_n)$, respectively.

This completes the proof of Claim 5. □

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Proof of Lemma 5. Putting Claims 3 to 5 together completes the proof of Lemma 5 as follows:

$$\begin{aligned}\mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}] &= (\text{Part I}) + (\text{Part II}) + (\text{Part III}) \\ &\in (\text{Part I}) \pm (|(\text{Part II})| + |(\text{Part III})|) \\ &= \rho_n \cdot (1 \pm o_n(1)).\end{aligned}$$

□

C.2 Upper Bound on the Second Moment of $\boldsymbol{\mathcal{Z}}$

Lemma 6 (Second Moment of $\boldsymbol{\mathcal{Z}}$)**.** *Fix a coefficient set $C = C(d) = \{\pm 1, \pm 2, \dots, \pm d\}$ for some integer $d > 1$, $M = O^*(|C|^{(1-\epsilon)n})$ for some constant $\epsilon > 0$, and a target $\tau = O(M)$. For ρ_n and f defined as in Equation (C.2) and Equation (C.3), we have*

$$\mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}^2] \le \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} |G(\theta_1, \theta_2)|^n \cdot \mathrm{d}\theta_1 \mathrm{d}\theta_2 \le \rho_n^2 \cdot (1 + o_n(1)).$$

where the function $G(\theta_1, \theta_2) := \sum_{(c_1,c_2)\in C^2} f(c_1\theta_1 + c_2\theta_2)$ for any $(\theta_1, \theta_2) \in [-\pi, \pi]^2$.

For any choice of $\vec{\boldsymbol{x}} \sim [0 : M-1]^n$, we deduce from (C.4) that

$$\begin{aligned}\boldsymbol{\mathcal{Z}}^2 &= \left(\frac{1}{2\pi}\int_{-\pi}^{\pi} \cos(\tau\theta) \prod_{j\in[n]} \Big(\sum_{c\in C} \cos(c\theta \boldsymbol{x}_j)\Big) \cdot \mathrm{d}\theta\right)^2 \\ &= \frac{1}{4\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \cos(t\theta_1)\cos(t\theta_2) \prod_{j\in[n]} \Big(\sum_{(c_1,c_2)\in C^2} \cos(c_1\theta_1\boldsymbol{x}_j)\cos(c_2\theta_2\boldsymbol{x}_j)\Big) \cdot \mathrm{d}\theta_1\mathrm{d}\theta_2 \\ &\le \frac{1}{4\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \prod_{j\in[n]} \left|\sum_{(c_1,c_2)\in C^2} \cos(c_1\theta_1\boldsymbol{x}_j)\cos(c_2\theta_2\boldsymbol{x}_j)\right| \cdot \mathrm{d}\theta_1\mathrm{d}\theta_2 \\ &= \frac{1}{4\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \prod_{j\in[n]} \left|\sum_{(c_1,c_2)\in C^2} \Big(\cos(c_1\theta_1\boldsymbol{x}_j)\cos(c_2\theta_2\boldsymbol{x}_j) - \sin(c_1\theta_1\boldsymbol{x}_j)\sin(c_2\theta_2\boldsymbol{x}_j)\Big)\right| \cdot \mathrm{d}\theta_1\mathrm{d}\theta_2 \\ &= \frac{1}{4\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \prod_{j\in[n]} \left|\sum_{(c_1,c_2)\in C^2} \cos(c_1\theta_1\boldsymbol{x}_j + c_2\theta_2\boldsymbol{x}_j)\right| \cdot \mathrm{d}\theta_1\mathrm{d}\theta_2,\end{aligned}$$

where the last second step holds since $C = C(d) = \{\pm 1, \pm 2, \dots, \pm d\}$ is a symmetric set and $\sin(z)$ is an odd function, and the last step uses the identity $\cos(\alpha+\beta) = \cos(\alpha)\cos(\beta) - \sin(\alpha)\sin(\beta)$.

Since all the $\boldsymbol{x}_j \sim [0 : M-1]$ are i.i.d., the following holds for the second moment $\mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}^2]$:

$$\begin{aligned}\mathbf{E}_{\vec{\boldsymbol{x}}}[\boldsymbol{\mathcal{Z}}^2] &\le \frac{1}{4\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \left|\sum_{(c_1,c_2)\in C^2} \mathop{\mathbf{E}}_{\boldsymbol{\xi}\sim[0:M-1]} \Big[\cos(c_1\theta_1\boldsymbol{\xi} + c_2\theta_2\boldsymbol{\xi})\Big]\right|^n \cdot \mathrm{d}\theta_1\mathrm{d}\theta_2 \\ &= \frac{1}{4\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \left|\sum_{(c_1,c_2)\in C^2} f(c_1\theta_1 + c_2\theta_2)\right|^n \cdot \mathrm{d}\theta_1\mathrm{d}\theta_2 \\ &= \frac{1}{4\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} |G(\theta_1, \theta_2)|^n \cdot \mathrm{d}\theta_1\mathrm{d}\theta_2, \qquad \text{(C.18)}\end{aligned}$$

where the second step uses the definition $f(\theta) = \mathbf{E}_{\boldsymbol{\xi}\sim[0:M-1]}[\cos(\theta\boldsymbol{\xi})]$ (see (C.3)).

Remark 2. We will show that the mass of (C.18) is concentrated around the origin, where $|G(\theta_1, \theta_2)|^n \approx |C|^{2n}$. Notably, this is *not* true when $C = C(1)$, as considered in [BCP01]. In the $C = C(1)$ case, mass is also concentrated at the points $[\pm\pi, \pm\pi]$, the corners of the region of integration. This is because the function $G(\theta_1, \theta_2) = \sum_{(c_1,c_2)\in C^2} f(c_1\theta_1 + c_2\theta_2)$ consists of four simple summands that interfere constructively at this point. However, when $C = C(d)$ for any $d > 1$, interference from other summands ensures that this phenomenon does not happen.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

We reuse the same constants a, b and b_0 (namely $|C|^2/\epsilon < a < b$ and $b_0 = 2M \cdot \sin^{-1}(\frac{b}{2M})$) introduced in the proof of Lemma 5. Define the subregions $R'_\theta \subseteq R''_\theta \subseteq [-\pi, \pi]^2$ by letting

$$R'_\theta := \Big[-\frac{\sqrt{64\ln(n)}}{M\sqrt{n}}, \frac{\sqrt{64\ln(n)}}{M\sqrt{n}} \Big]^2 \qquad \text{and} \qquad R''_\theta := \Big[-\frac{b_0}{M}, \frac{b_0}{M} \Big]^2.$$

Similar to the approach of [BCP01], we split the formula (C.18) into the following three parts.

$$\text{(Part 1)} \qquad \mathbf{E}_{\vec{x}}[\boldsymbol{\mathcal{Z}}] \le \frac{1}{4\pi^2} \iint_{(\theta_1,\theta_2)\in R'_\theta} |G(\theta_1,\theta_2)|^n \cdot \mathrm{d}\theta_1 \mathrm{d}\theta_2$$

$$\text{(Part 2)} \qquad + \frac{1}{4\pi^2} \iint_{(\theta_1,\theta_2)\in R''_\theta \setminus R'_\theta} |G(\theta_1,\theta_2)|^n \cdot \mathrm{d}\theta_1 \mathrm{d}\theta_2$$

$$\text{(Part 3)} \qquad + \frac{1}{4\pi^2} \iint_{(\theta_1,\theta_2)\in [-\pi,\,\pi]^2 \setminus R''_\theta} |G(\theta_1,\theta_2)|^n \cdot \mathrm{d}\theta_1 \mathrm{d}\theta_2.$$

We evaluate (Part 1) and (Part 2) respectively in Claims 6 and 7. (Part 3) is rather tricky; we first prove three auxiliary results (Claims 8 to 10) and then evaluate it in Corollary 5.

Claim 6. (Part 1) $= \rho_n^2 \cdot (1 \pm o_n(1))$.

Proof. (This proof is similar to the proof of Claim 3.) Following [BCP01], we set $y_1 := M\theta_1$ and $y_2 := M\theta_2$. Thus we consider small enough $|y_1|, |y_2| \le \sqrt{64\ln(n)/n} = o_n(1)$. In this range,

$$\begin{aligned} G(y_1/M, y_2/M) &= \sum_{(c_1,c_2)\in C^2} f(c_1 y_1/M + c_2 y_2/M) \\ &= \sum_{(c_1,c_2)\in C^2} \Big(1 - \frac{\kappa}{2}(c_1 y_1 + c_2 y_2)^2\Big) \cdot (1 \pm O(y_1^4 + y_2^4)) \\ &= \Big(|C|^2 - |C| \cdot \frac{\kappa \sum_{c\in C} c^2}{2} \cdot (y_1^2 + y_2^2)\Big) \cdot (1 \pm O(y_1^4 + y_2^4)) \\ &= |C|^2 \cdot \exp\Big(-\frac{\kappa \sum_{c\in C} c^2}{2|C|}(y_1^2 + y_2^2)\Big) \cdot (1 \pm O(y_1^4 + y_2^4)), \end{aligned}$$

where the first step changes variables $y_1 = M\theta_1$ and $y_2 = M\theta_2$, the second step holds since we have $f(y/M) = (1 - \frac{\kappa}{2}y^2) \cdot (1 \pm O(y^4))$ for small $|y|$ (see (C.7)), the third step is elementary algebra (notice that $C = C(d) = \{\pm 1, \pm 2, \ldots, \pm d\}$ is a symmetric set, so the crossing terms $2c_1c_2y_1y_2$ get cancelled), and the last step uses the approximation $e^{-z} = (1 - z) \cdot (1 \pm O(z^2))$ for small $|z|$.

As a consequence, the following holds for any $|y_1|, |y_2| \le \sqrt{64\ln(n)/n}$.

$$\begin{aligned} |G(y_1/M, y_2/M)|^n &= |C|^{2n} \cdot \exp\Big(-\frac{n\kappa \sum_{c\in C} c^2}{2|C|}(y_1^2 + y_2^2)\Big) \cdot (1 \pm O(n y_1^4 + n y_2^4)) \\ &= |C|^{2n} \cdot \exp\Big(-\frac{n\kappa \sum_{c\in C} c^2}{2|C|}(y_1^2 + y_2^2)\Big) \cdot (1 \pm o_n(1)). \end{aligned} \tag{C.19}$$

This allows us to bound the magnitude of (Part 1) as follows:

$$\begin{aligned} \text{(Part 1)} &= \frac{1}{4\pi^2 M^2} \iint_{|y_1|,|y_2|\le\sqrt{64\ln(n)/n}} |G(y_1/M, y_2/M)|^n \cdot \mathrm{d}y_1 \mathrm{d}y_2 \\ &= \frac{|C|^{2n}}{4\pi^2 M^2} \cdot (1 \pm o_n(1)) \cdot \iint_{|y_1|,|y_2|\le\sqrt{64\ln(n)/n}} \exp\Big(-\frac{n\kappa \sum_{c\in C} c^2}{2|C|}(y_1^2 + y_2^2)\Big) \cdot \mathrm{d}y_1 \mathrm{d}y_2 \\ &= \frac{|C|^{2n}}{4\pi^2 M^2} \cdot (1 \pm o_n(1)) \cdot \frac{2\pi|C|}{n\kappa \sum_{c\in C} c^2} \cdot \mathrm{erf}\left(\sqrt{\frac{32\kappa}{|C|}\Big(\sum_{c\in C} c^2\Big)\ln(n)} \right)^2 \end{aligned}$$

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

$$= \frac{|C|^{2n}}{4\pi^2 M^2} \cdot (1 \pm o_n(1)) \cdot \frac{2\pi|C|}{n\kappa \sum_{c\in C} c^2} \cdot (1 \pm o_n(1))^2$$
$$= \rho_n^2 \cdot (1 \pm o_n(1)).$$

Here the first step changes variables $y_1 = M\theta_1$ and $y_2 = M\theta_2$. The second step substitutes (C.19). The third step resolves the integral by using the Gaussian error function $\mathrm{erf}(z)$. The fourth step uses the approximation of $\mathrm{erf}(z)$ in (C.9), and the last step uses the definition of ρ_n (see (C.2)).

This completes the proof of Claim 6. □

Claim 7. (Part 2) $= o(\rho_n^2)$.

Proof. (This proof is similar to the proof of Claim 4.) Once again, we set $y_1 := M\theta_1$ and $y_2 := M\theta_2$. Then (Part 2) corresponds the range $(y_1, y_2) \in R''_y \setminus R'_y$, where

$$R'_y := [-\sqrt{64\ln(n)/n},\ \sqrt{64\ln(n)/n}]^2 \qquad \text{and} \qquad R''_y := [-b_0,\ b_0]^2.$$

For each pair $(c_1, c_2) \in C^2$, we define the function $G_{c_1,c_2}(y_1, y_2) := \frac{1}{2}(f(\frac{c_1 y_1}{M} + \frac{c_2 y_2}{M}) + f(\frac{c_2 y_1}{M} - \frac{c_1 y_2}{M}))$. We will show that for any pair $(c_1, c_2) \in C^2$ and any point $(y_1, y_2) \in R''_y \setminus R'_y$, we have

$$|G_{c_1,c_2}(y_1, y_2)| \le \frac{1}{\sqrt[n]{n^2}}. \tag{C.20}$$

Assuming (C.20) for the moment, for any point $(y_1, y_2) \in R''_y \setminus R'_y$ we have

$$\begin{aligned} |G(\theta_1, \theta_2)| = |G(y_1/M, y_2/M)| &= \left| \sum_{(c_1,c_2)\in C^2} f(c_1 y_1/M + c_2 y_2/M) \right| \\ &= \left| \sum_{(c_1,c_2)\in C^2} G_{c_1,c_2}(y_1, y_2) \right| \\ &\le \frac{|C|^2}{\sqrt[n]{n^2}}, \end{aligned} \tag{C.21}$$

where the second step holds since, regarding the definition of G_{c_1,c_2}, the mapping $(c_1, c_2) \mapsto (c_2, -c_1)$ is one-to-one, and the last step uses (C.20).

Hence, we can upper-bound (Part 2) as follows.

$$\begin{aligned} \text{(Part 2)} &= \frac{1}{4\pi^2 M^2} \iint_{y_1,y_2\in R''_y\setminus R'_y} |G(y_1/M + y_2/M)|^n \cdot \mathrm{d}y_1 \mathrm{d}y_2 \\ &\le \frac{1}{4\pi^2 M^2} \cdot |R''_y| \cdot \Big(\max_{y_1,y_2\in R''_y\setminus R'_y} |G(y_1/M + y_2/M)| \Big)^n \\ &\le \frac{1}{4\pi^2 M^2} \cdot |R''_y| \cdot \frac{|C|^{2n}}{n^2} \\ &= \frac{b_0^2 |C|^{2n}}{\pi^2 M^2 n^2} \\ &= o(\rho_n^2), \end{aligned}$$

where the first step changes variables $y_1 = M\theta_1$ and $y_2 = M\theta_2$, the third step substitutes (C.21), the fourth step is elementary algebra (notice that $|R''_y| = 4b_0^2$), and the last step holds because $b_0 \le \frac{\pi}{2} \cdot b$ is a constant (see (C.6)) and $\rho_n = \Theta(\frac{|C|^n}{M\sqrt{n}})$ (see (C.2)).

It remains to verify (C.20). Before doing so, we observe that for any point $(y_1, y_2) \in R''_y \setminus R'_y$,

$$\left| \frac{c_1 y_1 + c_2 y_2}{2M} \right| \le \frac{|c_1| + |c_2|}{2M} \cdot b_0 \le \frac{|c_1| + |c_2|}{16d \cdot b} \cdot (\pi/2) \cdot b \le \frac{|c_1| + |c_2|}{d} \cdot (\pi/4) \le \frac{\pi}{2}.$$

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

where the second step holds because $b_0 \le (\pi/2)\cdot b$ (see (C.6)) and $M \ge 8d\cdot b$ (see (C.5)), and the last step holds because $c \in C = C(d) = \{\pm 1, \pm 2, \ldots, \pm d\}$. Since we always have $|\frac{c_1y_1+c_2y_2}{2M}| \le \frac{\pi}{2}$ in the considered range, the arguments for (C.13) can be reused, resulting in

$$|f(c_1y_1/M + c_2y_2/M)| \le \Big|\frac{\sin(c_1y_1+c_2y_2)}{c_1y_1+c_2y_2}\Big| + \frac{1-\cos(c_1y_1+c_2y_2)}{40}. \tag{C.22}$$

Similarly, for any point $(y_1, y_2) \in R_y'' \setminus R_y'$, we also have $|\frac{c_2y_1-c_1y_2}{2M}| \le \frac{\pi}{2}$ and

$$|f(c_2y_1/M - c_1y_2/M)| \le \Big|\frac{\sin(c_2y_1-c_1y_2)}{c_2y_1-c_1y_2}\Big| + \frac{1-\cos(c_2y_1-c_1y_2)}{40}. \tag{C.23}$$

To prove (C.20) for any pair $(c_1, c_2) \in C^2$ and any point $(y_1, y_2) \in R_y'' \setminus R_y'$, let us do case analysis.

Case 1: When $|c_1y_1 + c_2y_2| \ge \frac{\pi}{2}$. We deduce that

$$\begin{aligned}
|G_{c_1,c_2}(y_1,y_2)| &= \frac{1}{2}\cdot\Big|f(c_1y_1/M + c_2y_2/M) + f(c_2y_1/M - c_1y_2/M)\Big| \\
&\le \frac{1}{2}\cdot\Big(\text{RHS of (C.22)} + 1\Big) \\
&\le \frac{1}{2}\cdot\Big(\frac{2}{\pi} + \frac{1}{20} + 1\Big) \\
&\le \frac{1}{\sqrt[n]{n^2}},
\end{aligned}$$

where the second step uses (C.22) and the fact that $|f(\theta)| \le 1$ for any $\theta \in \mathbb{R}$ (see (C.3)), the third step holds since $|\frac{\sin(z)}{z}| + \frac{1-\cos(z)}{40} \le \frac{2}{\pi} + \frac{1}{20}$ when $|z| \ge \frac{\pi}{2}$ (see the proof of Claim 4, **Case II**), and the last step holds since $\frac{1}{2}\cdot(\frac{2}{\pi} + \frac{1}{20} + 1) \approx 0.8433$ and $\frac{1}{\sqrt[n]{n^2}} = 1 - o_n(1)$.

Case 2: When $|c_2y_1 - c_1y_2| \ge \frac{\pi}{2}$. Here we can reapply the arguments for **Case 1**.

Case 3: When $|c_1y_1 + c_2y_2| \le \frac{\pi}{2}$ and $|c_2y_1 - c_1y_2| \le \frac{\pi}{2}$. Combining (C.22) and (C.23) gives

$$\begin{aligned}
|G_{c_1,c_2}(y_1,y_2)| &= \frac{1}{2}\cdot\Big|f(c_1y_1/M + c_2y_2/M) + f(c_2y_1/M - c_1y_2/M)\Big| \\
&\le \frac{1}{2}\cdot\Big(\text{RHS of (C.22)} + \text{RHS of (C.23)}\Big) \\
&\le \frac{1}{2}\cdot\Big(e^{-\frac{1}{8}(c_1y_1+c_2y_2)^2} + e^{-\frac{1}{8}(c_2y_1-c_1y_2)^2}\Big),
\end{aligned} \tag{C.24}$$

where the third step holds since $|\frac{\sin(z)}{z}| + \frac{1-\cos(z)}{40} \le e^{-z^2/8}$ for any $|z| \le \frac{\pi}{2}$ (see the proof of Claim 4, **Case I**; notice that $|\frac{c_1y_1+c_2y_2}{2M}| \le \frac{\pi}{2}$ and $|\frac{c_2y_1-c_1y_2}{2M}| \le \frac{\pi}{2}$).

Let us give a geometric interpretation for (C.24), which is demonstrated in Figure 3. Consider two straight lines $L' : c_1z_1 + c_2z_2 = 0$ and $L'' : c_2z_1 - c_1z_2 = 0$. By elementary algebra, we can see that the Euclidean distance $d' := d'(y_1, y_2)$ (resp. the Euclidean distance $d'' := d''(y_1, y_2)$) from a given point $(y_1, y_2) \in \mathbb{R}$ to straight line L' (resp. straight line L'') is given by

$$d' = \frac{|c_1y_1 + c_2y_2|}{\sqrt{c_1^2 + c_2^2}} \qquad \text{and} \qquad d'' = \frac{|c_2y_1 - c_1y_2|}{\sqrt{c_1^2 + c_2^2}}. \tag{C.25}$$

Moreover, L' and L'' are perpendicular (by construction), so the Pythagorean theorem ensures that $d'^2 + d''^2 = y_1^2 + y_2^2$. Accordingly, the larger distance $\max\{d',\ d''\}$ is lower bounded by

$$\max\{d',\ d''\} \ge \sqrt{(d'^2 + d''^2)/2} = \sqrt{(y_1^2 + y_1^2)/2} \ge \sqrt{32\ln(n)/n}, \tag{C.26}$$

where the first step uses the AM-GM inequality, and the last step holds since $(y_1, y_2) \notin R_y' = [-\sqrt{64\ln(n)/n},\ \sqrt{64\ln(n)/n}]^2$ (see Figure 3).

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

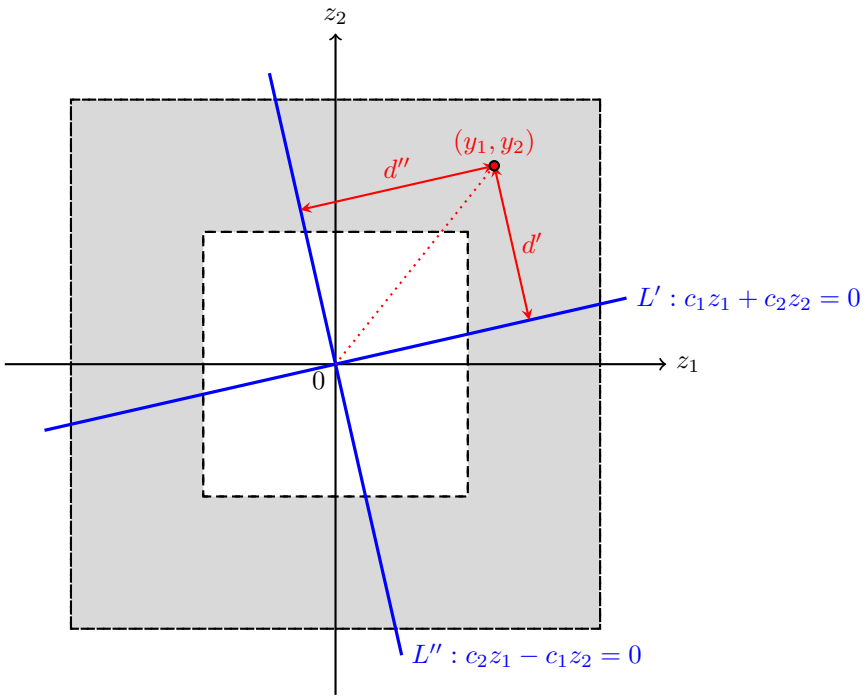


Figure 3: Demonstration for the formula (C.24) $\geq |G_{c_1,c_2}(y_1, y_2)|$. The gray region represents the considered range $(y_1, y_2) \in R''_y \setminus R'_y$, where $R'_y = [-\sqrt{64\ln(n)/n},\ \sqrt{64\ln(n)/n}]^2$ and $R''_y = [-b_0,\ b_0]^2$. The Pythagorean theorem guarantees that $d'^2 + d''^2 = y_1^2 + y_2^2$.

Putting everything together results in

$$\begin{aligned}
|G_{c_1,c_2}(y_1, y_2)| \leq (\text{C.24}) &= \frac{1}{2} \cdot \left(e^{-\frac{c_1^2+c_2^2}{8} \cdot d'^2} + e^{-\frac{c_1^2+c_2^2}{8} \cdot d''^2} \right) \\
&\leq \frac{1}{2} \cdot \left(e^{-\frac{1}{4} \cdot d'^2} + e^{-\frac{1}{4} \cdot d''^2} \right) \\
&\leq \frac{1}{2} \cdot \left(e^{-\frac{1}{4} \cdot 0} + e^{-\frac{1}{4} \cdot 32\ln(n)/n} \right) \\
&= \frac{1}{2} \cdot \left(1 + \frac{1}{\sqrt[n]{n^8}} \right) \\
&\leq \frac{1}{\sqrt[n]{n^2}},
\end{aligned}$$

where the first step applies (C.25), the second step follows since the coefficients $c_1, c_2 \in C$ are nonzero integers, the third step applies (C.26) and the fact that $\min\{d',\ d''\} \geq 0$, the fourth step is elementary algebra, and the last step holds whenever $n \geq 6$ (see (C.5)).

Combining all the three cases together gives (C.20). This completes the proof of Claim 7. □

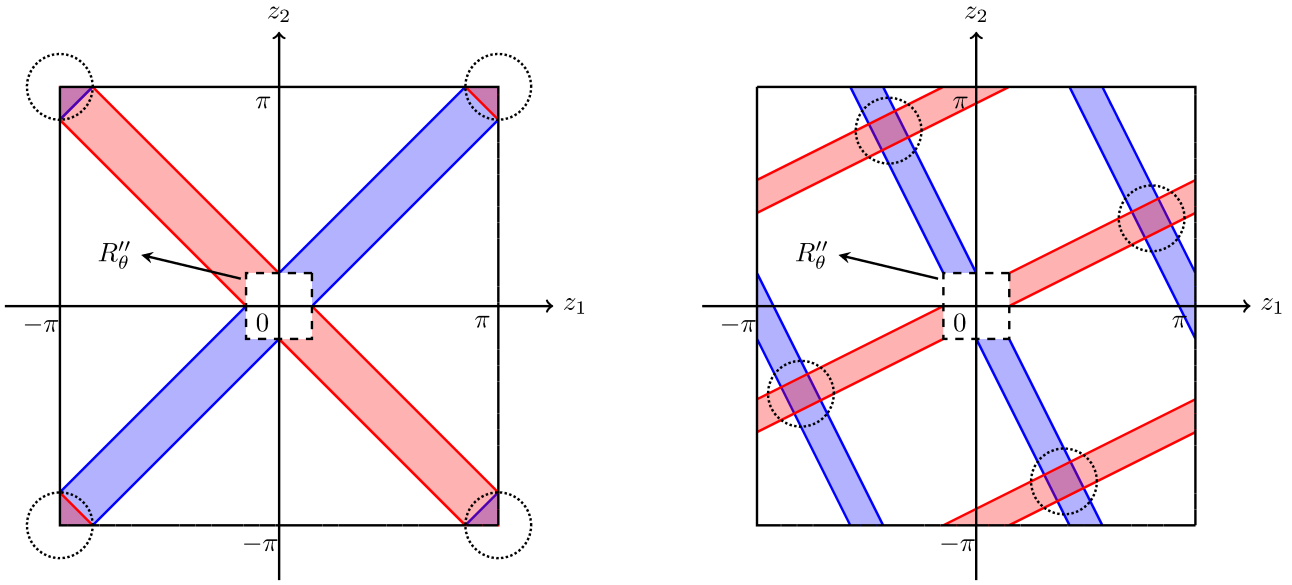
Below we use Claims 8 to 10 (as auxiliaries) to evaluate (Part 3) in Corollary 5. This evaluation together with Claims 6 and 7 immediately gives Lemma 6.

Claim 8. *There exists some constant $\epsilon_2 := \epsilon_2(|C|, a) > 0$ such that, for any $(\theta_1, \theta_2) \in [-\pi,\ \pi]^2 \setminus R''_\theta$,*

$$|G(\theta_1, \theta_2))| \leq |C|^2 - \frac{1 - 1/a}{2} \leq |C|^{(2-\epsilon_2)}.$$

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy



(a) When $(c_1, c_2) = (1, 1)$ (b) When $(c_1, c_2) = (2, 1)$

Figure 4: The subregion $R_{c_1,c_2} := \{(\theta_1, \theta_2) \in [-\pi, \pi]^2 : (c_1\theta_1 + c_2\theta_2), (c_2\theta_1 - c_1\theta_2) \in [-\frac{b_0}{M}, \frac{b_0}{M}]_\bigcirc\}$ is the intersection of the red area (on which $(c_1\theta_1 + c_2\theta_2) \in [-\frac{b_0}{M}, \frac{b_0}{M}]_\bigcirc$) and the blue area (on which $(c_2\theta_1 - c_1\theta_2) \in [-\frac{b_0}{M}, \frac{b_0}{M}]_\bigcirc$). Given any parameters $t_R, t_B \in \mathbb{R}$, by construction a (red) straight line $c_1\theta_1 + c_2\theta_2 = t_R$ and a (blue) straight line $c_2\theta_1 - c_1\theta_2 = t_B$ are vertical. The subregion R''_θ (namely the dashed square) is excluded from $[-\pi, \pi]^2$. Figures 4a and 4b respectively choose $(c_1, c_2) = (1, 1)$ and $(c_1, c_2) = (2, 1)$.

Proof. Define the function $f_{c_1,c_2}(\theta_1, \theta_2) := \frac{1}{2}(f(c_1\theta_1 + c_2\theta_2) + f(c_2\theta_1 - c_1\theta_2))$ for each pair $(c_1, c_2) \in C^2$. Because $|f(\theta)| \le 1$ for any $\theta \in \mathbb{R}$ (see (C.3)), we also have $|f_{c_1,c_2}(\theta_1, \theta_2)| \le 1$ for any $(\theta_1, \theta_2) \in \mathbb{R}^2$. Recall that $|f(\theta)| \le \frac{1}{a}$ for any $|\theta| \in [\frac{b_0}{M}, \pi]$ (see (C.14)) and $\frac{b_0}{M} \le \frac{(\pi/2)\cdot b}{8d\cdot b} \le \frac{\pi}{16}$ (see (C.5) and (C.6)). Since f is 2π-periodic (see (C.3)), only on the 2π-periodic range $[-\frac{b_0}{M}, \frac{b_0}{M}]_\bigcirc := \bigcup_{k\in\mathbb{Z}}[-\frac{b_0}{M} + 2k\pi, \frac{b_0}{M} + 2k\pi]$ can we possibly have $|f(\theta)| > \frac{1}{a}$. Regarding the function f_{c_1,c_2}, we consider the analogous subregion

$$R_{c_1,c_2} := \Big\{(\theta_1, \theta_2) \in [-\pi, \pi]^2 : (c_1\theta_1 + c_2\theta_2), (c_2\theta_1 - c_1\theta_2) \in [-\tfrac{b_0}{M}, \tfrac{b_0}{M}]_\bigcirc\Big\}.$$

For any point $(\theta_1, \theta_2) \in [-\pi, \pi]^2 \setminus R_{c_1,c_2}$, we have either $|f(c_1\theta_1 + c_2\theta_2)| \le \frac{1}{a}$ or $|f(c_2\theta_1 - c_1\theta_2)| \le \frac{1}{a}$ or both, which (together with the fact that $|f(\theta)| \le 1$ for any $\theta \in \mathbb{R}$) implies

$$|f_{c_1,c_2}(\theta_1, \theta_2)| \le \frac{|f(c_1\theta_1 + c_2\theta_2)| + |f(c_2\theta_1 - c_1\theta_2)|}{2} \le \frac{1 + 1/a}{2}. \tag{C.27}$$

Because $C = C(d) = \{\pm 1, \pm 2, \ldots, \pm d\}$ for a fixed integer $d > 1$, the considered pair-indices $(c_1, c_2) \in C^2$ include both $(c_1, c_2) = (1, 1)$ and $(c_1, c_2) = (2, 1)$. As shown in Figure 4a, the union of the $\frac{b_0}{M}$-neighborhoods of four points $(\pm\pi, \pm\pi)$ covers $R_{1,1}$. Also, as shown in Figure 4b, the union of the $\frac{b_0}{M}$-neighborhoods of four points $(\frac{4\pi}{5}, \frac{2\pi}{5})$, $(-\frac{2\pi}{5}, \frac{4\pi}{5})$, $(-\frac{4\pi}{5}, -\frac{2\pi}{5})$ and $(\frac{2\pi}{5}, -\frac{4\pi}{5})$ covers $R_{2,1}$. We must have $R_{1,1} \cap R_{2,1} = \emptyset$, because the $\frac{b_0}{M} \le \frac{\pi}{16}$ is small enough.

Given the above arguments, we deduce that for any $(\theta_1, \theta_2) \in [-\pi, \pi]^2 \setminus R''_\theta$,

$$\begin{aligned} |G(\theta_1, \theta_2)| &= \left| \sum_{(c_1,c_2)\in C^2} f(c_1\theta_1 + c_2\theta_2) \right| \\ &= \left| \sum_{(c_1,c_2)\in C^2} f_{c_1,c_2}(\theta_1, \theta_2) \right| \end{aligned}$$

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

$$\leq \frac{1+1/a}{2} + |C|^2 - 1$$
$$= |C|^2 - \frac{1-1/a}{2}.$$

Here the second step holds since, regarding the definition of f_{c_1,c_2}, the mapping $(c_1, c_2) \mapsto (c_2, -c_1)$ is one-to-one. The third step uses (C.27), the fact that $|f_{c_1,c_2}(\theta_1, \theta_2)| \leq 1$ for each pair $(c_1, c_2) \in C^2$ and any $(\theta_1, \theta_2) \in \mathbb{R}^2$, and the fact that $R_{1,1} \cap R_{2,1} = \emptyset$.

Because $a > |C|^2/\epsilon > 1$ (see (C.5)), there must exist some constant $\epsilon_2 := \epsilon_2(|C|, a) > 0$ such that $|C|^2 - (1 - 1/a)/2 \leq |C|^{(2-\epsilon_2)}$. This finishes the proof of Claim 8. □

Claim 9. *The subregion $Q' \subseteq [-\pi, \pi]^2$ defined below has measure at most $|Q'| \leq |C|^2 \cdot \frac{4\pi b_0}{M}$.*

$$Q' := \Big\{(\theta_1, \theta_2) \in [-\pi, \pi]^2 : |G(\theta_1, \theta_2)| > \frac{|C|^2}{a}\Big\}.$$

Proof. To begin with, we introduce an auxiliary set $A_{c_1,w}$ for each $c_1 \in C$ and any offset $w \in \mathbb{R}$.

$$A_{c_1,w} := \Big\{\theta_1 \in [-\pi, \pi] : |f(c_1\theta_1 + w)| > \frac{1}{a}\Big\}. \tag{C.28}$$

Recall (C.16) that $|A_{c_1,0}| \leq \frac{2b_0}{M}$. Because f is 2π-periodic, regarding any integer $c_1 \in C$, the function $f_{c_1}(\theta_1) := f(c_1\theta_1)$ is also 2π-periodic. Thus, we also have

$$|A_{c_1,w}| \leq \frac{2b_0}{M} \tag{C.29}$$

for any $c_1 \in C$, $w \in \mathbb{R}$. It follows that, for any pair $(c_1, c_2) \in C^2$, we have

$$\Big|\Big\{(\theta_1, \theta_2) \in [-\pi, \pi]^2 : |f(c_1\theta_1 + c_2\theta_2)| > \frac{1}{a}\Big\}\Big| \leq 2\pi \cdot \frac{2b_0}{M} = \frac{4\pi b_0}{M}. \tag{C.30}$$

Recall that $G(\theta_1, \theta_2) = \sum_{(c_1,c_2)\in C^2} f(c_1\theta_1 + c_2\theta_2)$. Suppose $|G(\theta_1, \theta_2)| > |C|^2/a$, then we must have $|f(c_1\theta_1 + c_2\theta_2)| > 1/a$ for at least one pair $(c_1, c_2) \in C^2$. Union-bounding the measure (C.30) over all pairs $(c_1, c_2) \in C^2$ concludes the proof of Claim 9. □

Claim 10. *The subregion $Q'' \subseteq [-\pi, \pi]^2$ defined below has measure at most $|Q''| \leq |C|^4 \cdot (\frac{2b_0}{M})^2$.*

$$Q'' := \Big\{(\theta_1, \theta_2) \in [-\pi, \pi]^2 : |G(\theta_1, \theta_2)| > |C| + \epsilon\Big\}.$$

Proof. For this proof, we introduce a convenient decomposition of the function G. Observe that $G(\theta_1, \theta_2) = \sum_{c_2\in C} q(\theta_1, c_2\theta_2)$, where

$$q(\theta_1, w) := \sum_{c_1\in C} f(c_1\theta_1 + w).$$

This allows us to rewrite Q'' as

$$Q'' = \Big\{(\theta_1, \theta_2) \in [-\pi, \pi]^2 : \Big|\sum_{c_2\in C} q(\theta_1, c_2\theta_2)\Big| > |C| + \epsilon\Big\}.$$

As $\max_x q(x) \leq |C| \cdot \max_x f(x) \leq |C|$, q is upper-bounded by $|C|$. Thus for any point $(\theta_1, \theta_2) \in Q''$, there must exist two distinct coefficients $c_2, c_2' \in C$ such that $|q(\theta_1, c_2\theta_2)|, |q(\theta_1, c_2'\theta_2)| > \frac{\epsilon}{|C|}$. In other words, the considered set Q'' is covered by

$$Q'' \subseteq \bigcup_{c_2\neq c_2'\in C} Q_{c_2,c_2'}, \tag{C.31}$$

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

where

$$Q_{\alpha,\beta} := \Big\{(\theta_1,\theta_2) \in [-\pi,\pi]^2 : |q(\theta_1,\alpha\theta_2)|, |q(\theta_1,\beta\theta_2)| \geq \frac{|C|}{a}\Big\}.$$

This definition of $Q_{\alpha,\beta}$ suffices because $a > |C|^2/\epsilon$ (see (C.5)) and thus $\frac{|C|}{a} < \frac{\epsilon}{|C|}$.

To complete the proof, it suffices to show that

$$|Q_{c_2,c_2'}| \leq |C|^2 \cdot \Big(\frac{2b_0}{M}\Big)^2 \tag{C.32}$$

for each pair $(c_2, c_2') \in C^2$ with $c_2 \neq c_2'$. (Suppose so, then the measure $|Q''|$ can be upper-bounded by $|C|^2 \cdot \max_{c_2 \neq c_2'} |Q_{c_2,c_2'}| \leq |C|^4 \cdot (\frac{2b_0}{M})^2$ using (C.31).)

To show (C.32), fix a pair $(c_2, c_2') \in C^2$ such that $c_2 \neq c_2'$. We observe that

$$Q_{c_2,c_2'} = \Big\{(\theta_1,\theta_2) \in [-\pi,\pi]^2 : \theta_1 \in \big(B_{c_2\theta_2} \cap B_{c_2'\theta_2}\big)\Big\},$$

for $B_w := \{\theta_1 \in [-\pi,\pi] : |q(\theta_1,w)| > \frac{|C|}{a}\}$. Intuitively, B_w captures the values for which $|q(\theta_1,w)|$ is large. Because B_w is determined by the function $q(\theta_1,w) = \sum_{c\in C} f(c_1\theta_1 + w)$, it has a convenient property: for any $w, z \in \mathbb{R}$, B_{w+z} is just the set B_w translated by $z \pmod{2\pi}$. Figure 5 provides a visual aid for the structure of this set and the remainder of the proof.

It will be convenient to write $|Q_{c_2,c_2'}|$ in terms of the probability that a random point in $[-\pi,\pi]^2$ is contained in $Q_{c_2,c_2'}$. We have

$$\begin{aligned}
|Q_{c_2,c_2'}| &= 4\pi^2 \cdot \mathop{\mathbf{Pr}}_{\boldsymbol{\theta_1},\boldsymbol{\theta_2}\sim[-\pi,\pi]}\Big[\boldsymbol{\theta}_1 \in \big(B_{c_2\boldsymbol{\theta}_2} \cap B_{c_2'\boldsymbol{\theta}_2}\big)\Big] \\
&= 4\pi^2 \cdot \mathop{\mathbf{Pr}}_{\boldsymbol{\theta_1},\boldsymbol{\theta_2}\sim[-\pi,\pi]}\Big[\boldsymbol{\theta}_1 \in \big(B_0 \cap B_{(c_2'-c_2)\boldsymbol{\theta}_2}\big)\Big] \\
&= 4\pi^2 \cdot \mathop{\mathbf{Pr}}_{\boldsymbol{\theta}_1\sim[-\pi,\pi]}\Big[\boldsymbol{\theta}_1 \in B_0\Big] \cdot \mathop{\mathbf{Pr}}_{\boldsymbol{\theta_1},\boldsymbol{\theta_2}\sim[-\pi,\pi]}\Big[\boldsymbol{\theta}_1 \in B_{(c_2'-c_2)\boldsymbol{\theta}_2} \;\Big|\; \boldsymbol{\theta}_1 \in B_0\Big].
\end{aligned} \tag{C.33}$$

Here the second step follows from translating both sets by $c_2\boldsymbol{\theta}_2 \pmod{2\pi}$, which does not change the size of the intersection. And the last step uses the identity $\mathbf{Pr}[A \cap B] = \mathbf{Pr}[A] \cdot \mathbf{Pr}[B \mid A]$.

Given any nonzero integer k and any fixed offset $w \in [-\pi,\pi]$, we observe that

$$\mathop{\mathbf{Pr}}_{\boldsymbol{\theta}_2\sim[-\pi,\pi]}[w \in B_{k\boldsymbol{\theta}_2}] = \mathop{\mathbf{Pr}}_{\boldsymbol{y}\sim[-\pi,\pi]}[\boldsymbol{y} \in B_0] = \frac{|B_0|}{2\pi}, \tag{C.34}$$

because sampling $\boldsymbol{\theta_2}$ uniformly from $[-\pi,\pi]$ distributes $k\boldsymbol{\theta}_2 \pmod{2\pi}$ uniformly over $[-\pi,\pi]$. (C.34) holds for any nonzero integer $k \neq 0$ and any offset $w \in [-\pi,\pi]$, so both (conditional) probabilities in (C.33) can be substituted with (C.34) $= |B_0|/(2\pi)$.

Recall the definition of the set family $A_{c_1,w}$ in (C.28). Using (C.29) to union-bound (C.33) yields (C.32):

$$|Q_{c_2,c_2'}| = 4\pi^2 \cdot \Big(\frac{|B_0|}{2\pi}\Big)^2 \leq \Big(\sum_{c\in C} |A_{c,0}|\Big)^2 \leq |C|^2 \cdot \Big(\frac{2b_0}{M}\Big)^2. \qquad \square$$

Corollary 5. (Part 3) $= o(\rho_n^2)$.

Proof. For ease of reference, let us restate the results in Claims 8 to 10:

- Claim 8: $|G(\theta_1,\theta_2))| \leq |C|^{(2-\epsilon_2)}$ for any point $(\theta_1,\theta_2) \in [-\pi,\ \pi]^2 \setminus R_\theta''$.
- Claim 9: The subregion Q' on which $|G(\theta_1,\theta_2)| > |C|^2/a$ has measure at most $|C|^2 \cdot \frac{4\pi b_0}{M}$.
- Claim 10: The subregion Q'' on which $|G(\theta_1,\theta_2)| > |C| + \epsilon$ has measure at most $|C|^4 \cdot (\frac{2b_0}{M})^2$.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

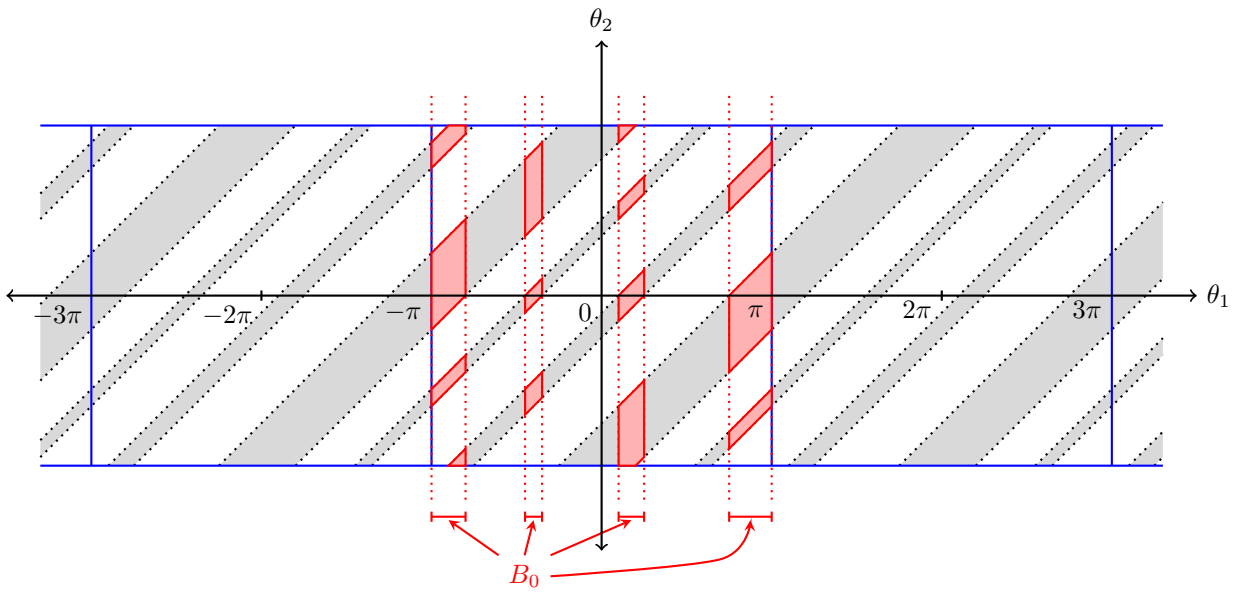


Figure 5: Visual aid for the proof of Claim 10. For any offset $\theta_2 \in [-\pi, \pi]$, the range $B_{(c_2'-c_2)\theta_2}$ is the intersection of the line segment $(-\pi, \theta_2)$–(π, θ_2) with the gray regions. The red regions indicate $\{(\theta_1, \theta_2) \in [-\pi, \pi]^2 : \theta_1 \in B_0 \cap B_{(c_2'-c_2)\theta_2}\}$.

Obviously, the three subregions $(Q'' \setminus R_\theta'')$ and $(Q' \setminus Q'')$ and $[-\pi,\ \pi]^2 \setminus Q'$ together covers the domain of integration for (Part 3), namely $[-\pi,\ \pi]^2 \setminus R_\theta''$. As a consequence, we have

$$\begin{aligned}
\text{(Part 3)} &\le \frac{1}{4\pi^2}\Bigg(\iint_{(\theta_1,\theta_2)\in[-\pi,\ \pi]^2\setminus Q'} + \iint_{(\theta_1,\theta_2)\in Q'\setminus Q''} + \iint_{(\theta_1,\theta_2)\in Q''\setminus R_\theta''}\Bigg)|G(\theta_1,\theta_2)|^n \cdot \mathrm{d}\theta_1\mathrm{d}\theta_2 \\
&\le \frac{1}{4\pi^2}\left(\left(\frac{|C|^2}{a}\right)^n \cdot 4\pi^2 + (|C|+\epsilon)^n \cdot |Q'| + |C|^{(2-\epsilon_2)n} \cdot |Q''|\right) \\
&\le \frac{1}{4\pi^2}\left(\left(\frac{|C|^2}{a}\right)^n \cdot 4\pi^2 + (|C|+\epsilon)^n \cdot |C|^2 \cdot \frac{4\pi b_0}{M} + |C|^{(2-\epsilon_2)n} \cdot |C|^4 \cdot \left(\frac{2b_0}{M}\right)^2\right) \\
&= O(\epsilon^n) + O\Big(\frac{(|C|+\epsilon)^n}{M}\Big) + O\Big(\frac{|C|^{(2-\epsilon_2)n}}{M^2}\Big).
\end{aligned}$$

Here the second step follows from Claim 8 and the definitions of Q' and Q''. The third step applies Claims 9 and 10. The fourth step holds since (for the first term) $a > |C|^2/\epsilon$ using (C.5) and (for the second/third terms) both $|C|$ and b_0 are constants. Since $M = O^*(|C|^{(1-\epsilon)n})$ for a constant $\epsilon > 0$, we know from (C.2) that $\rho_n^2 = \Theta(|C|^{2n}/(M^2 n)) = \Omega(|C|^{\epsilon n})$. Observe that:

- The first term $O(\epsilon^n) = o_n(1) = o(\rho_n^2)$.
- The second term $O((|C|+\epsilon)^n \cdot M^{-1}) = O(\rho_n^2) \cdot (|C|+\epsilon)^n \cdot \frac{Mn}{|C|^{2n}}$, by the definition of ρ_n (C.2). Then an upper bound $O(\rho_n^2) \cdot |C|^{-(\epsilon/4)n} = o(\rho_n^2)$ can be inferred from the calculation below.

$$\begin{aligned}
(|C|+\epsilon)^n \cdot \frac{Mn}{|C|^{2n}} &= O\Big((|C|+\epsilon)^n \cdot \frac{1}{|C|^{(1+\epsilon/2)n}}\Big) \\
&= O\Big(|C|^{(1+\epsilon/4)n} \cdot \frac{1}{|C|^{(1+\epsilon/2)n}}\Big) \\
&= O(|C|^{-(\epsilon/4)n}).
\end{aligned}$$

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Here the first step holds because $M = O^*(|C|^{(1-\epsilon)n})$ and then (given that $\epsilon > 0$ is a constant) $Mn = O(|C|^{(1-\epsilon/2)n})$. The second step holds since $|C| + \epsilon \le |C| \cdot (1 + \epsilon/4) < |C|^{1+\epsilon/4}$, given that $|C| \ge 4$ (namely $C = C(d) = \{\pm 1, \dots, \pm d\}$ for some integer $d > 1$).

- The third term $O(|C|^{(2-\epsilon_2)n} \cdot M^{-2}) = O(\rho_n^2) \cdot n \cdot |C|^{-\epsilon_2 n} = o(\rho_n^2)$, given that $\epsilon_2 = \epsilon_2(|C|, n) > 0$ is a constant (see the statement of Claim 8).

Putting everything together completes the proof of Corollary 5. □

Proof of Lemma 6. Putting Claims 6 and 7 and Corollary 5 together, we have

$$\mathbf{E}_{\vec{x}}[\boldsymbol{\mathcal{Z}}^2] \le (\text{Part 1}) + (\text{Part 2}) + (\text{Part 3}) \le \rho_n \cdot (1 + o_n(1)).$$

This finishes the proof of Lemma 6. □

Downloaded 04/24/23 to 160.39.60.240 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited