

# Improving Lagarias-Odlyzko Algorithm for Average-Case Subset Sum: Modular Arithmetic Approach

Antoine Joux  

CISPA Helmholtz Center for Information Security, Saarbrücken, Germany

Karol Węgrzycki  

Max Planck Institute for Informatics, Saarbrücken, Germany

---

## Abstract

Lagarias and Odlyzko (J.ACM 1985) proposed a polynomial-time algorithm for solving “almost all” instances of the Subset Sum problem with  $n$  integers of size  $\Omega(\Gamma_{LO})$ , where  $\log_2(\Gamma_{LO}) > n^2 \log_2(\gamma)$  and  $\gamma$  is a parameter of the lattice basis reduction ( $\gamma > \sqrt{4/3}$  for LLL). The algorithm of Lagarias and Odlyzko is a cornerstone of cryptography. However, the theoretical guarantee on the density of feasible instances has remained unimproved for almost 40 years.

In this paper, we propose an algorithm that solves “almost all” instances of Subset Sum with integers of size  $\Omega(\sqrt{\Gamma_{LO}})$  after a single call to lattice reduction. Additionally, our approach allows solving the Subset Sum problem for multiple targets, whereas the previous method could handle only one target per call to lattice basis reduction. We introduce a modular arithmetic approach to the Subset Sum problem, leveraging lattice reduction to solve a linear system modulo a suitably large prime. By analyzing the lengths of the LLL-reduced basis vectors of both the primal and dual lattices simultaneously, we show that density guarantees can be improved.

**2012 ACM Subject Classification** Security and privacy → Cryptanalysis and other attacks; Theory of computation → Randomness, geometry and discrete structures

**Keywords and phrases** Average-Case Analysis, Subset Sum, Lattice Reduction, LLL

**Digital Object Identifier** 10.4230/LIPIcs.STACS.2026.57

**Related Version** *Full Version*: <https://arxiv.org/abs/2408.16108> [35]

**Funding** *Karol Węgrzycki*: Supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) grant number 559177164.

## 1 Introduction

The Subset Sum problem, a fundamental problem of computational complexity and cryptography, has been extensively studied due to its numerous practical implications. In the Subset Sum problem, we are given  $n$  positive integers  $a_1, \dots, a_n \in \mathbb{Z}$  and a target  $T \in \mathbb{Z}$ . The task is to decide if there exists a subset of the given numbers that sum to  $T$ .

In particular, the study of Subset Sum has seen numerous applications in cryptography [41, 23], balancing [20], and combinatorial optimization [36]. With these applications in mind, our goal is to propose an efficient algorithm to solve the Subset Sum for a large range of instances. Indeed, since the problem is NP-hard, it is unlikely that we could propose an efficient algorithm to solve all instances of the problem. The natural next step is to characterize which instances of the problem can be solved efficiently. On that front, there is a classical result by Bellman [11] that proposes a polynomial time algorithm for Subset Sum when the numbers are polynomially bounded, i.e., the instance is *dense*. This result inspired a long line of pseudopolynomial-time algorithms (see e.g., [17, 37, 45]).



© Antoine Joux and Karol Węgrzycki;

licensed under Creative Commons License CC-BY 4.0

43rd International Symposium on Theoretical Aspects of Computer Science (STACS 2026).

Editors: Meena Mahajan, Florin Manea, Annabelle McIver, and Nguyễn Kim Thăng

Article No. 57; pp. 57:1–57:19



Leibniz International Proceedings in Informatics

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



At the other end of the spectrum, Lagarias and Odlyzko [38] introduced a polynomial time algorithm for *sparse* instances. Their work marked a significant milestone by proposing a polynomial-time algorithm capable of solving “almost all” instances of Subset Sum when the numbers are of size  $\Omega(\Gamma_{LO})$ , where  $\log_2(\Gamma_{LO}) > n^2 \log_2 \gamma$  and  $\gamma$  is a parameter associated with the lattice reduction. Lagarias and Odlyzko [38] employed the LLL (Lenstra-Lenstra-Lovász [40]) algorithm for lattice basis reduction, for which  $\gamma = \gamma_{LLL} > \sqrt{4/3}$ .

To this day, the Lagarias and Odlyzko algorithm remains a cornerstone of modern cryptography and is featured in numerous text-books on the field [26, 15, 43]. This algorithm was subsequently implemented [46, 49, 16], simplified [24] and was improved assuming access to an exact Closest Vector Problem (CVP) [22, 34]. Nevertheless, the algorithm of Lagarias and Odlyzko is currently the only known method<sup>1</sup> to solve low-density instances of Subset Sum, both in theory and practice (see [44] for survey).

In this paper, we revisit and improve the result of Lagarias and Odlyzko [38]. We present an algorithm solving almost all instances of Subset Sum when the size of the numbers is  $\Omega(\sqrt{\Gamma_{LO}})$ . More formally, we prove the following theorem.

► **Theorem 1.** *Let  $a_1, \dots, a_n \in \mathbb{Z}_{\geq 0}$  be integers, selected uniformly at random from  $\{1, \dots, \lceil \sqrt{\Gamma_{LO}} \rceil\}$ , where  $\Gamma_{LO} = (\gamma_{LLL})^{n^2} \cdot 2^{\Theta(n \log n)}$ . With overwhelming probability, in polynomial time and after a single call to a basis reduction algorithm with parameter  $\gamma_{LLL}$ , we obtain a polynomial time tester, that for every given  $T \in \mathbb{Z}$  decides whether*

$$\text{there exist } e_1, \dots, e_n \in \{0, 1\} \text{ such that } \sum_{i=1}^n e_i \cdot a_i = T.$$

*Our tester and its creation method are both deterministic. Their success probability only depends on the random choice of the integers  $a_1, \dots, a_n$  and is at least  $1 - 2^{-\Omega(n \log n)}$ .*

The improvement offered by Theorem 1 is twofold. First, in terms of the admissible range, our algorithm improves the bound on the instance’s numbers from  $\Omega(\Gamma_{LO})$  down to  $\Omega(\sqrt{\Gamma_{LO}})$ . This means that, within the same running time as the previous algorithm, we can solve more instances of Subset Sum (see discussion in Section 1.2). Second, our algorithm is *oblivious* to the target. Namely, after a single call to the lattice basis reduction, with input numbers  $a_1, \dots, a_n$  we can deterministically decide on an answer in polynomial time. In this sense, our argument provides a succinct certificate for “almost all” instances of sparse Subset Sum. Precisely this question was previously considered by Furst and Kannan [25] who used lattice basis reduction to bound the proof complexity of Subset Sum. They strengthened the Lagarias and Odlyzko algorithm and showed that “almost all” instances with numbers in  $\{1, \dots, \Gamma_{LO}\}$  have a polynomial size proof. Our argument again reduces the range to  $\{1, \dots, \lceil \sqrt{\Gamma_{LO}} \rceil\}$ , and our certificates are simpler. In fact, as we will see later, the data-structure behind Theorem 1 requires only a single matrix-vector multiplication.

**Running time comparison.** Lagarias and Odlyzko offer a reduction of the average-case Subset Sum problem to a lattice problem. The exact parameters for which the Subset Sum can be solved depend on the quality of the lattice reduction algorithm used. In their original approach, they performed this reduction using a perfect lattice reduction and the LLL algorithm. They assumed access to an algorithm that provided an approximate shortest vector within an approximation factor of  $\gamma^n$ .

<sup>1</sup> There are several extensions of the LLL algorithm tailored for solving Subset Sum in practice, see [49, 39].

Theorem 1 provides a more efficient, in terms of the running time, reduction than that of Lagarias and Odlyzko. For simplicity of exposition, we first prove Theorem 1, which uses the LLL algorithm as a subroutine. Later, in Section 1.2, we extend this result to a family of lattice reduction algorithms where the parameter  $\gamma \rightarrow 1$ . Although polynomial, these algorithms incur a higher running time cost as  $\gamma$  approaches 1. In particular, for any fixed constant  $c > 0$ , average-case instances of Subset Sum with numbers in the range  $\{1, \dots, \Gamma_c\}$ , where

$$\log_2(\Gamma_c) = \frac{n^2 \log \log n}{c \log n},$$

can be solved in  $n^{c+\mathcal{O}(1)}$  time using the Lagarias and Odlyzko approach. This also holds for our approach, and in fact, we achieve the same density using a lattice reduction algorithm with a running time  $n^{c/2+\mathcal{O}(1)}$ , offering improved efficiency. We provide further details in Section 1.2 and Appendix A.

### 1.1 Our Technique: Modular Arithmetic Approach

Now, we elaborate on the techniques that we introduce. In this note, we let  $\llbracket a \rrbracket_p$  be a unique integer in  $(-p/2, p/2)$  such that  $a \equiv \llbracket a \rrbracket_p \pmod{p}$  (see Section 2 for notation). Consider vectors  $\mathbf{a} = (a_1, \dots, a_n) \in \mathbb{Z}^n$ ,  $\mathbf{e} = (e_1, \dots, e_n) \in \{0, 1\}^n$  and an integer  $T$  such that

$$e_1 \cdot a_1 + e_2 \cdot a_2 + \dots + e_n \cdot a_n = T.$$

Obviously, for any integers  $\mu, p \in \mathbb{Z}$  the following also holds

$$e_1 \cdot \llbracket \mu a_1 \rrbracket_p + e_2 \cdot \llbracket \mu a_2 \rrbracket_p + \dots + e_n \cdot \llbracket \mu a_n \rrbracket_p \equiv \llbracket \mu T \rrbracket_p \pmod{p}.$$

Next, assume, that we have managed to find  $\mu$  and  $p$ , such that:

$$\sum_{i=1}^n |\llbracket \mu a_i \rrbracket_p| < p/2.$$

Then there are no overflows modulo  $p$  and actually, we can write

$$e_1 \cdot \llbracket \mu a_1 \rrbracket_p + e_2 \cdot \llbracket \mu a_2 \rrbracket_p + \dots + e_n \cdot \llbracket \mu a_n \rrbracket_p = \llbracket \mu T \rrbracket_p.$$

We will show that if the density is small, then there are integers  $\mu_1, \dots, \mu_n \in \mathbb{Z}$  such that for each of them (i) the  $\ell_1$  norm is bounded, i.e.,  $\|\llbracket \mu_i \mathbf{a} \rrbracket_p\|_1 < p/2$ , and (ii) the vectors  $\llbracket \mu_i \mathbf{a} \rrbracket_p$  are linearly independent. More formally we will prove the following theorem.

► **Theorem 2 (Modular Arithmetic Approach).** *Let  $\Gamma_{LO} := \Theta((\gamma_{LLL})^{n^2} \cdot 2^{4n \log_2 n})$  and let  $p = \Theta((\gamma_{LLL})^{0.5n^2})$  be a prime. There exists a polynomial time algorithm, that given integers  $a_1, \dots, a_n \in \mathbb{Z}$  selected uniformly at random from  $\{1, \dots, \lceil \sqrt{\Gamma_{LO}} \rceil\}$  outputs, with overwhelming probability, integers  $\mu_1, \dots, \mu_n \in \mathbb{Z}_p$  such that the following  $n \times n$  matrix*

$$\mathcal{M}_p := \begin{pmatrix} \llbracket \mu_1 a_1 \rrbracket_p & \llbracket \mu_1 a_2 \rrbracket_p & \cdots & \llbracket \mu_1 a_n \rrbracket_p \\ \llbracket \mu_2 a_1 \rrbracket_p & \llbracket \mu_2 a_2 \rrbracket_p & \cdots & \llbracket \mu_2 a_n \rrbracket_p \\ \vdots & & & \vdots \\ \llbracket \mu_n a_1 \rrbracket_p & \llbracket \mu_n a_2 \rrbracket_p & \cdots & \llbracket \mu_n a_n \rrbracket_p \end{pmatrix}$$

has full rank. Moreover, for every  $1 \leq i \leq n$  it holds that:

$$\|\llbracket \mu_i a_1 \rrbracket_p, \dots, \llbracket \mu_i a_n \rrbracket_p\|_1 < \frac{p}{2}.$$

More precisely, the procedure succeeds with probability  $\geq 1 - 2^{-\Omega(n \log n)}$ , which only depends on the random choice of integers  $a_1, \dots, a_n$ .

The algorithm for Theorem 1 follows easily with the matrix  $\mathcal{M}_p$  in hand. First we let  $\mathbf{t}_p := (\llbracket \mu_1 T \rrbracket_p, \dots, \llbracket \mu_n T \rrbracket_p)^\top$  and compute the inverse of  $\mathcal{M}_p$  (which is possible because  $\mathcal{M}_p$  has full-rank). It remains to verify that the vector  $(\mathcal{M}_p)^{-1} \cdot \mathbf{t}$  is in  $\{0, 1\}^n$  (see Section 3 for the full proof).

Observe that if  $\mathbf{e} \in \{0, 1\}^n$  is a solution for  $\langle \mathbf{a}, \mathbf{e} \rangle = T$ , then  $\mathcal{M}_p \cdot \mathbf{e}^\top = \mathbf{t}$ . Therefore, if a solution in  $\{0, 1\}^n$  exists, we find it. On the other hand, recall, that  $\mathcal{M}_p$  has full rank, hence  $\mathbf{e}$  such that  $\mathcal{M}_p \cdot \mathbf{e}^\top = \mathbf{t}$  is *unique*. This means that, if we find  $\mathbf{e}$  that satisfies  $\mathcal{M}_p \cdot \mathbf{e}^\top = \mathbf{t}$ , yet  $\mathbf{e}$  is not in  $\{0, 1\}^n$  we know that no solution exists. When we find a  $\{0, 1\}^n$  solution, we also check that it is indeed a genuine solution of the Subset Sum instance.

Now, we briefly sketch the proof of Theorem 2. Our algorithm is inspired by Howgrave-Graham's revisit [30] of Coppersmith's method for finding short roots of univariate modular equations [21]. In a nutshell, we construct a lattice (very similar to the original lattice of Lagarias and Odlyzko) for Subset Sum modulo a prime number  $p$ . Analogously to the original approach of Lagarias and Odlyzko we bound the length of vectors in this lattice. The caveat is that we can also bound the lengths of the vectors in the *dual* lattice. This crucially allows us to double the density of the admissible instances.

There are two main differences between our proposal and Howgrave-Graham's algorithm [30] (and the original Coppersmith's method [21]) Firstly, the number of variables to consider in our case is much larger. Secondly, we only need to construct linear polynomials in these variables rather than considering large degree polynomials.

## 1.2 Extension to Block Basis Reduction

We would like to comment on the possible improvements of using stronger lattice reduction. Specifically, the only aspect of the Lagarias-Odlyzko algorithm that relies on the LLL algorithm is the fact that it provides a  $(\gamma_{\text{LLL}})^n$ -approximate solution to the Shortest Vector Problem with  $\gamma_{\text{LLL}} > \sqrt{4/3}$ . Instead of the LLL algorithm, it is natural to apply block-basis reduction (e.g., [48, 2]) with parameter  $\Upsilon$  arbitrarily close to 1. This way, one expects a better approximation of SVP, consequently improving the range  $\Gamma$ . The obvious drawback of this is the increased running time.

Our approach, in principle, should allow us to reduce the bitlengths of the Subset Sum instances that can be solved regardless of the concrete algorithm behind the lattice reduction. Unfortunately, adapting our approach to work with every basis lattice reduction requires a separate analysis which is much more technical than for LLL. Nevertheless, the takeaway message is as follows:

The Modular Arithmetic Approach allows us to solve almost all instances of Subset Sum with integers  $\Omega(\sqrt{\Gamma})$  with a single call to lattice reduction, whereas the previous approach guaranteed to solve only instances with integers  $\Omega(\Gamma)$  while using the same lattice reduction algorithm.

Since the LLL algorithm is the most popular we decided first to present our algorithms using this algorithm. In Appendix A, we demonstrate that our approach works for the textbook block-reduction presented by Gama and Nguyen [27, 43].

► **Theorem 3.** *Let  $\Upsilon > 1$  be a parameter of lattice reduction in [27]. Let  $a_1, \dots, a_n \in \mathbb{Z}$  be positive integers, selected uniformly at random from range  $\{1, \dots, \lceil \sqrt{\Gamma} \rceil\}$ , where  $\Gamma = \Upsilon^{n^2} \cdot 2^{\Theta(n \log n)}$ . With overwhelming probability, after a single call to a basis reduction*

algorithm in [27] with parameter  $\Upsilon$ , we obtain a polynomial time tester, that for every given  $T \in \mathbb{Z}$  decides whether

$$\text{there exist } e_1, \dots, e_n \in \{0, 1\} \text{ such that } \sum_{i=1}^n e_i \cdot a_i = T,$$

Our tester and its creation method are both deterministic. Their success probability only depends on the random choice of the integers  $a_1, \dots, a_n$  and is at least  $1 - 2^{-\Omega(n \log n)}$ .

In comparison, the standard analysis of the Lagarias-Odlyzko algorithm requires the range of integers to be of order  $\Gamma = \Upsilon^{n^2} \cdot 2^{\Theta(n \log n)}$  when the lattice reduction with parameter  $\Upsilon > 1$  is used.

### 1.3 Related Work

The currently fastest algorithm to solve Subset Sum exactly in the worst-case settings runs in  $\mathcal{O}^*(2^{n/2})$  time [29]. Schroepel and Shamir [50] show that Subset Sum admits a time-space tradeoff, i.e., an algorithm using  $\mathcal{S}$  space and  $2^n/\mathcal{S}^2$  time for any  $\mathcal{S} \leq \mathcal{O}^*(2^{n/4})$ . This tradeoff was improved by [4] for almost all tradeoff parameters (see also [23, 42]).

Average-case algorithms for Subset Sum and more general problems are actively researched [33, 13, 14, 10, 12, 51]. When input numbers are of order  $2^n$ , Howgrave-Graham and Joux [31] showed  $\mathcal{O}^*(2^{0.337n})$  time and  $\mathcal{O}^*(2^{0.256n})$  space algorithm for Subset Sum. This was subsequently improved by Becker, Coron and Joux [9] to  $\mathcal{O}^*(2^{0.291n})$  time and space.

The exact running time of the Subset Sum was also analyzed in the parameterized setting [5, 6], quantum regime [3, 28], total regime [1, 19, 7] and the polynomial space regime [8]. From the perspective of pseudopolynomial algorithms, Subset Sum has also been the subject of recent stimulative research [17, 45, 32, 18].

## 2 Preliminaries

Throughout the paper, we let  $[n] := \{1, \dots, n\}$ . We use  $\equiv_p$  to denote equivalence modulo  $p$ , i.e.,  $a \equiv_p b$  iff there exists  $k \in \mathbb{Z}$  such that  $a = k \cdot p + b$ . For integers  $a, b$  we let  $\llbracket a \rrbracket_b$  be the unique value in set  $\{-\lceil b/2 \rceil + 1, \dots, \lfloor b/2 \rfloor\}$  such that  $\llbracket a \rrbracket_b \equiv_b a$ .  $\mathcal{O}^*(\cdot)$  notation hides factors polynomial in  $n$ .

We consider  $\mathbb{R}^n$  with the usual Euclidean topology. We use bold letters to denote vectors. The inner product of two vectors  $\mathbf{x} = (x_i)_{i=1}^n$  and  $\mathbf{y} = (y_i)_{i=1}^n$  is denoted by  $\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=1}^n x_i \cdot y_i$ . We denote the  $\ell_p$  norm as  $\|\mathbf{x}\|_p := (\sum_{i=1}^n (x_i)^p)^{1/p}$ . Often, we omit the subscript  $p$  which means that we consider the Euclidean case with  $p = 2$ . We denote the  $d$ -dimensional Euclidean ball of radius  $R$  centered at  $\mathbf{0}$  by  $\text{Ball}_d(R)$ . We use the Stirling formula to approximate the volume of the  $d$ -dimensional ball, namely:

$$\text{Vol}(\text{Ball}_d(R)) \sim \frac{1}{\sqrt{d\pi}} \left( \frac{2\pi e}{d} \right)^{d/2} \cdot R^d = 2^{-\Theta(d \log(d))} \cdot R^d.$$

### 2.1 Background on Lattices

A *lattice* in  $\mathbb{R}^n$  is a discrete subgroup of  $(\mathbb{R}^n, +)$ . This implies that a lattice is a non-empty set  $L \subseteq \mathbb{R}^n$  such that for all  $\mathbf{x}, \mathbf{y} \in L$  it holds that  $\mathbf{x} - \mathbf{y} \in L$ . Any subgroup of  $(\mathbb{Z}^n, +)$  is a lattice, and it is called an *integral lattice*.

Let  $\mathbf{b}_1, \dots, \mathbf{b}_m$  be vectors in  $\mathbb{R}^n$ . Let  $\mathcal{L}(\mathbf{b}_1, \dots, \mathbf{b}_m)$  be the set of integral combinations of these vectors, namely:

$$\mathcal{L}(\mathbf{b}_1, \dots, \mathbf{b}_m) := \left\{ \sum_{i=1}^m \ell_i \cdot \mathbf{b}_i \text{ such that } \ell_1, \dots, \ell_m \in \mathbb{Z} \right\}.$$

Then,  $\mathcal{L}(\mathbf{b}_1, \dots, \mathbf{b}_m)$  is a lattice, specifically the lattice spanned by  $\mathbf{b}_1, \dots, \mathbf{b}_m$ . Furthermore, when the vectors  $\mathbf{b}_1, \dots, \mathbf{b}_m$  are linearly independent (over  $\mathbb{R}$ ), we say that they form a basis of  $\mathcal{L}(\mathbf{b}_1, \dots, \mathbf{b}_m)$ . In this case, every vector  $\mathbf{v} \in L$  uniquely decomposes as an integral combination of the basis vectors, i.e., for every  $\mathbf{v} \in L$  there exists unique  $\alpha_1, \dots, \alpha_m \in \mathbb{Z}$  such that  $\mathbf{v} = \sum_{i=1}^m \alpha_i \cdot \mathbf{b}_i$  [43, Definition 5, p. 26]. It is well-known that every lattice admits a basis. The cardinality of any basis is an invariant of the lattice and is called its *dimension* (or sometimes referred to as its rank). This quantity, denoted by  $\dim(L)$  is equal to the dimension of its linear span (denoted  $\text{span}(L)$ ) as a vector space over  $\mathbb{R}$ .

We say that a lattice  $L \subseteq \mathbb{R}^n$  has full rank when  $n = \dim(L)$ . Since any lattice of  $\mathbb{R}^n$  admits some basis, we assume w.l.o.g. that every lattice we consider is given in the form  $\mathcal{L}(\mathbf{b}_1, \dots, \mathbf{b}_m)$  for some  $\mathbf{b}_i$ 's in  $\mathbb{R}^n$  where  $m = \dim(L)$ .

Thus,  $\mathbf{b}_1, \dots, \mathbf{b}_m \in \mathbb{R}^n$  are linearly independent vectors. Their Gram-Schmidt orthogonalization (GSO) is the orthogonal family  $(\mathbf{b}_1^*, \dots, \mathbf{b}_m^*)$  defined inductively as follows:

$$\mathbf{b}_1^* = \mathbf{b}_1 \quad \text{and}$$

$$\mathbf{b}_i^* = \mathbf{b}_i - \sum_{j=1}^{i-1} \mu_{i,j} \mathbf{b}_j^* \quad \text{where} \quad \mu_{i,j} := \frac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\|\mathbf{b}_j^*\|^2} \quad \text{for all } 1 \leq j < i \leq m.$$

We can define the volume of the lattice  $L$  spanned by  $\mathbf{b}_1, \dots, \mathbf{b}_m$  as

$$\text{Vol}(L) = \prod_{i=1}^m \|\mathbf{b}_i^*\|,$$

and remark that it is a lattice invariant and does not depend on the choice of basis.

Other classical invariants of a lattice are its successive minima. By definition, the  $i$ th successive minimum, denoted as  $\lambda_i(L)$  is the smallest real number for which there exist  $i$  linearly independent vectors in  $L$  each of length at most  $\lambda_i(L)$ . In particular,  $\lambda_1(L)$  is the length of the shortest nonzero vector in  $L$ . Note that  $\lambda_1(L) \leq \lambda_2(L) \leq \dots \leq \lambda_m(L)$ .

## 2.2 Dual Lattice

For a lattice  $L \subset \mathbb{R}^n$  the *dual* lattice of  $L$  is defined as:

$$L^\dagger := \{\mathbf{y} \in \text{span}(L) \text{ such that } \langle \mathbf{x}, \mathbf{y} \rangle \in \mathbb{Z} \text{ for all } \mathbf{x} \in L\}.$$

When  $L$  is an  $m$ -dimensional lattice of  $\mathbb{R}^n$  then  $L^\dagger$  is also an  $m$ -dimensional lattice of  $\mathbb{R}^n$ . More precisely, if  $L$  is spanned by the rows of the basis matrix  $B$ , then  $L^\dagger$  is spanned by the rows of:

$$B^\dagger = (BB^\top)^{-1} B$$

Furthermore, it holds that  $\text{Vol}(L) \cdot \text{Vol}(L^\dagger) = 1$ . Finally, if we reverse the order of the rows of  $B^\dagger$ , we obtain the so-called dual basis  $D^\dagger$  of  $B$ . The GSO of dual basis is related to the GSO of the primal basis by the following relations:

$$\|\mathbf{b}_i^*\| \cdot \|\mathbf{d}_{m-i+1}^*\| = 1 \quad \text{for every } i \in [m].$$

In order to make explicit that the order of the (row) basis vectors is reversed, we write:

$$D^\dagger = \mathbf{Rev} \left( (BB^\top)^{-1} B \right).$$

### 2.3 The LLL algorithm and its property

Lenstra, Lenstra and Lovász [40] presented a polynomial time procedure that, given an arbitrary basis of a lattice computes a so-called reduced basis with better properties. Extensions of LLL can also compute a reduced basis from an arbitrary generating family.

The main parameter of LLL is a real  $\delta \in (0.25, 1)$ . A secondary parameter  $\mu \in [0.5, 1)$  is also useful when considering implementations of LLL with finite precision on orthogonalised vectors. The initial LLL with exact rational arithmetic [40] uses  $\mu = 1/2$ . Let  $\mathbf{b}_1, \dots, \mathbf{b}_m \in \mathbb{R}^n$  be basis of lattice  $L$  and  $\mathbf{b}_1^*, \dots, \mathbf{b}_m^* \in \mathbb{R}^n$  be its Gram-Schmidt orthogonalized basis of  $L$ . We say that  $(\mathbf{b}_1, \dots, \mathbf{b}_m)$  is  $(\delta, \mu)$ -LLL reduced if and only if the following conditions are satisfied:

$$\begin{aligned} \text{Size Reduction:} \quad & |\mu_{i,k}| \leq \mu && \text{for every } 1 \leq k < i \leq m \\ \text{Lovász Condition:} \quad & \delta \|\mathbf{b}_i^*\|^2 \leq \|\mathbf{b}_{i+1}^*\|^2 + \mu_{i+1,i}^2 \|\mathbf{b}_i^*\|^2 && \text{for every } 1 \leq i < m \end{aligned}$$

Recall that  $\mu_{i,k} = \frac{\langle \mathbf{b}_i, \mathbf{b}_k^* \rangle}{\langle \mathbf{b}_k^*, \mathbf{b}_k^* \rangle}$ . Both of these conditions yield an *implied parameter*  $\gamma_{\text{LLL}} := \frac{1}{\sqrt{\delta - \mu^2}}$ , such that:

$$\|\mathbf{b}_i^*\| \leq \gamma_{\text{LLL}} \cdot \|\mathbf{b}_{i+1}^*\| \quad \text{for every } 1 \leq i < m \tag{1}$$

Equation (1) is the strongest for  $\gamma_{\text{LLL}} = \sqrt{4/3} + \varepsilon$  for some  $\varepsilon > 0$ . This is achieved by taking  $\mu$  sufficiently close to  $1/2$  and  $\delta$  sufficiently close to  $1$ . In the sequel, we focus on the parameter  $\gamma_{\text{LLL}}$  and express our results in terms of it. Note, that the same inequality holds for the dual lattice.

The running time of LLL algorithm [40] is  $\mathcal{O}(m^5 n \cdot \text{polylog}(B))$ , where  $B$  is the Euclidean length of the longest basis vector) [40, 43] (see [47] for recent development).

### 2.4 State-of-the-art: Lagarias-Odlyzko Algorithm

Assume that  $\mathbf{a} = (a_1, \dots, a_n)$  is a vector in  $\mathbb{Z}^n$  where each  $a_i$  is chosen independently uniformly at random from  $\{1, \dots, \Gamma\}$  and  $\mathbf{e} = (e_1, \dots, e_n) \in \{0, 1\}^n$  is chosen independently at random. Set  $T = \langle \mathbf{a}, \mathbf{e} \rangle$ . Then, clearly  $\mathbf{e}$  is a solution to the equation:

$$\sum_{i=1}^n a_i x_i = T \text{ such that } x_i \in \{0, 1\} \text{ for every } i \in \{1, \dots, n\}.$$

Here, we recall the argument of Lagarias and Odlyzko who show that if  $\Gamma \geq \Omega((\gamma_{\text{LLL}})^{n^2 + o(n^2)})$  and  $\gamma_{\text{LLL}} > \sqrt{4/3}$ , then the LLL algorithm is sufficient to solve an instance  $a_1, \dots, a_n$  with target  $T$  of Subset Sum in polynomial time and with probability  $\geq 1 - 2^{-\Omega(n \log n)}$ .

Note that  $T \leq \sum_{i=1}^n a_i$ , and we can assume that  $\|\mathbf{e}\|_1 \leq \frac{n}{2}$  as otherwise, we can consider the same instance with target  $(\sum_{i=1}^n a_i - T)$ . Let  $K := \lceil n(\gamma_{\text{LLL}})^n \rceil$  be a sufficiently large integer. Lagarias and Odlyzko [38] construct the following row-generated lattice:

$$L := \begin{pmatrix} 1 & 0 & \cdots & 0 & a_1 \cdot K \\ 0 & 1 & \cdots & 0 & a_2 \cdot K \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & a_n \cdot K \\ 0 & 0 & \cdots & 0 & T \cdot K \end{pmatrix}.$$

## 57:8 Improving Lagarias-Odlyzko Algorithm

Let  $\bar{\mathbf{e}} = (e_1, \dots, e_n, 0)$  such that  $\sum_{i=1}^n e_i a_i = T$ . Observe that  $\bar{\mathbf{e}} \in L$  and  $\|\bar{\mathbf{e}}\| \leq \sqrt{n/2}$ . Therefore, the vector  $\bar{\mathbf{e}}$  is a *short vector* of  $L$ .

The idea of Lagarias and Odlyzko is to use the algorithm of Lenstra, Lenstra and Lovász [40]. Let  $\mathbf{v}$  be a minimum length non-zero vector in  $L$ . The LLL algorithm guarantees finding a nonzero vector  $\mathbf{x} \in L$  with:

$$\|\mathbf{x}\| \leq (\gamma_{\text{LLL}})^n \|\mathbf{v}\|.$$

Here,  $\gamma_{\text{LLL}}$  is an implied parameter of the LLL algorithm, and we can guarantee that when  $\gamma_{\text{LLL}} > \sqrt{4/3}$ , the LLL algorithm runs in polynomial time (see Section 2). Note that the vector  $\mathbf{x}$  found by the LLL algorithm satisfies

$$\|\mathbf{x}\| \leq (\gamma_{\text{LLL}})^n \|\bar{\mathbf{e}}\| \leq (\gamma_{\text{LLL}})^n \sqrt{n/2} < K$$

The strategy is, hence, to run the LLL algorithm and hope that the vector  $\mathbf{x}$  returned by LLL is either  $\bar{\mathbf{e}}$  or  $-\bar{\mathbf{e}}$ . There is of course the possibility that there are other spurious vectors of length less than  $K$ . The analysis by Lagarias and Odlyzko demonstrates that this is, however, highly unlikely. More precisely, they show the following statement:

▷ **Claim 4** (cf., [24]).  $\Pr$  [exists  $\mathbf{x} \in L \setminus \{k \cdot \bar{\mathbf{e}} \mid k \in \mathbb{Z}\}$  such that  $\|\mathbf{x}\| < K$ ]  $\leq \frac{(2K+1)^{n+1}}{\Gamma}$ .

Note that this is  $2^{-\Omega(n \log n)}$  when  $\Gamma \geq (\gamma_{\text{LLL}})^{n^2+o(n^2)}$ . Therefore, by considering the opposite event, we can guarantee that the vector  $\bar{\mathbf{e}}$  is found with a probability at least  $1 - 2^{-\Omega(n \log n)}$ . We include the proof of Claim 4 due to [24] as it serves as an introduction to our technique.

*Proof of Claim 4.* Let  $\mathbf{w} = (w_1, \dots, w_{n+1}) \in L$ . Observe that if  $w_{n+1} \neq 0$  then already  $\|\mathbf{w}\| \geq K$ . Hence, it must hold that  $w_{n+1} = 0$ . Let

$$\mathcal{W} := \{\mathbf{w} \in L \text{ such that } \|\mathbf{w}\| < K \text{ and } w_{n+1} = 0 \text{ and } \mathbf{w} \neq k \cdot \bar{\mathbf{e}} \text{ for any } k \in \mathbb{Z}\}.$$

Now, it suffices to bound the probability that  $\mathcal{W}$  is empty. If, however,  $\mathcal{W}$  is nonempty, there exists  $\mathbf{w} = (w_1, \dots, w_n, 0) \in \mathbb{Z}^{n+1}$  and  $\ell \in \mathbb{Z}$  that satisfy  $\mathbf{w} \neq \ell \cdot \mathbf{e}$  and:

$$\sum_{i=1}^n a_i \cdot w_i = \ell \cdot T. \tag{2}$$

Let us now fix this vector  $\mathbf{w}$  and integer  $\ell$ . Let  $\mathbf{z} = (z_1, \dots, z_n) \in \mathbb{Z}^n$  be such that  $z_i = w_i - e_i \cdot \ell$ . Observe that  $\langle \mathbf{a}, \mathbf{z} \rangle = 0$ . Without loss of generality, we can assume that  $z_1 \neq 0$  and we let  $Z := -(\sum_{i=2}^n a_i z_i / z_1)$ . Hence,

$$\Pr [\langle \mathbf{a}, \mathbf{z} \rangle = 0] = \Pr [a_1 = Z] = \sum_{i=1}^{\Gamma} \Pr [a_1 = i \mid Z = i] \cdot \Pr [Z = i].$$

As  $a_1$  and  $Z$  are independent, this is bounded by

$$\sum_{i=1}^{\Gamma} \frac{1}{\Gamma} \cdot \Pr [Z = i] \leq \frac{1}{\Gamma}.$$

Therefore, a fixed  $\mathbf{w}$  and  $\ell$  satisfy Equation (2) with probability at most  $1/\Gamma$ . Note that the number of  $\mathbf{w} \in \mathbb{Z}^{n+1}$  and  $\ell \in \mathbb{Z}$  such that  $\|\mathbf{w}\| < K$ ,  $w_{n+1} = 0$  and  $|\ell| \leq K$  is at most  $(2K+1)^{n+1}$  and the proof concludes. ◁

Note, that the proof of Claim 4 uses lattice reduction as a blackbox. If one were to use lattice reduction with parameter  $\Upsilon > 1$ , then the range guaranteed by Claim 4 would be  $\Gamma = \Theta((\Upsilon)^{n^2+o(n^2)})$ . In the next sections, we will show a method to improve the admissible range down to  $\Gamma = \Theta((\Upsilon)^{0.5n^2+o(n^2)})$  that uses more properties of lattice reduction. In Section 3 we analyse it with the standard LLL. Then in section Appendix A we analyse it with a textbook block lattice reduction of Gama and Nguyen [27].

### 3 Modular arithmetic approach

In this section, we focus on the proof of the following statement.

► **Theorem 2 (Modular Arithmetic Approach).** *Let  $\Gamma_{LO} := \Theta((\gamma_{LLL})^{n^2} \cdot 2^{4n \log_2 n})$  and let  $p = \Theta((\gamma_{LLL})^{0.5n^2})$  be a prime. There exists a polynomial time algorithm, that given integers  $a_1, \dots, a_n \in \mathbb{Z}$  selected uniformly at random from  $\{1, \dots, \lceil \sqrt{\Gamma_{LO}} \rceil\}$  outputs, with overwhelming probability, integers  $\mu_1, \dots, \mu_n \in \mathbb{Z}_p$  such that the following  $n \times n$  matrix*

$$\mathcal{M}_p := \begin{pmatrix} \llbracket \mu_1 a_1 \rrbracket_p & \llbracket \mu_1 a_2 \rrbracket_p & \cdots & \llbracket \mu_1 a_n \rrbracket_p \\ \llbracket \mu_2 a_1 \rrbracket_p & \llbracket \mu_2 a_2 \rrbracket_p & \cdots & \llbracket \mu_2 a_n \rrbracket_p \\ \vdots & & & \vdots \\ \llbracket \mu_n a_1 \rrbracket_p & \llbracket \mu_n a_2 \rrbracket_p & \cdots & \llbracket \mu_n a_n \rrbracket_p \end{pmatrix}$$

has full rank. Moreover, for every  $1 \leq i \leq n$  it holds that:

$$\|(\llbracket \mu_i a_1 \rrbracket_p, \dots, \llbracket \mu_i a_n \rrbracket_p)\|_1 < \frac{p}{2}.$$

More precisely, the procedure succeeds with probability  $\geq 1 - 2^{-\Omega(n \log n)}$ , which only depends on the random choice of integers  $a_1, \dots, a_n$ .

Before we do so, let us show how to use Theorem 2 to solve the Subset Sum formally.

**Proof of Theorem 1 assuming Theorem 2.** We start by describing the algorithm. First, we use Theorem 2 to compute the integers  $\mu_1, \dots, \mu_n, p$ . This gives us the matrix  $\mathcal{M}_p$ . We know that this matrix has full rank so we can compute its inverse with the Gaussian elimination algorithm. This concludes the description of the preprocessing phase.

Now, given a target  $T$  we compute the vector  $\mathbf{t}_p := (\llbracket \mu_1 T \rrbracket_p, \dots, \llbracket \mu_n T \rrbracket_p)^\top$  and compute a candidate solution  $\mathbf{e}^\top = \mathcal{M}_p^{-1} \cdot \mathbf{t}_p$ . Finally, we check that  $\mathbf{e}$  is indeed correct. Namely, if  $\sum_{i=1}^n e_i a_i = T$  and  $\mathbf{e} \in \{0, 1\}^n$  we return  $\mathbf{e}$  and  $\perp$  otherwise. This concludes the description of the algorithm. Clearly, the algorithm runs in polynomial time. Moreover, a query takes only  $\mathcal{O}(n^2)$  arithmetic operations (on numbers bounded by  $\lceil \sqrt{\Gamma_{LO}} \rceil$ ). The success probability of the algorithm comes exclusively from a single application of Theorem 2. Therefore, it remains to argue about the correctness.

Let  $e_1, \dots, e_n \in \{0, 1\}^n$  be a solution to  $\sum_{i=1}^n a_i e_i = T$  and assume that our algorithm returns  $\perp$ . Because the norm  $\|(\llbracket \mu_i \mathbf{a} \rrbracket_p)\|_1$  is bounded by  $p/2$  it holds that:

$$\begin{pmatrix} \llbracket \mu_1 a_1 \rrbracket_p & \llbracket \mu_1 a_2 \rrbracket_p & \cdots & \llbracket \mu_1 a_n \rrbracket_p \\ \llbracket \mu_2 a_1 \rrbracket_p & \llbracket \mu_2 a_2 \rrbracket_p & \cdots & \llbracket \mu_2 a_n \rrbracket_p \\ \vdots & & & \vdots \\ \llbracket \mu_n a_1 \rrbracket_p & \llbracket \mu_n a_2 \rrbracket_p & \cdots & \llbracket \mu_n a_n \rrbracket_p \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{pmatrix} = \begin{pmatrix} \llbracket \mu_1 T \rrbracket_p \\ \llbracket \mu_2 T \rrbracket_p \\ \vdots \\ \llbracket \mu_n T \rrbracket_p \end{pmatrix}$$

Since the matrix  $\mathcal{M}_p$  has full rank, there exists a unique solution to the above linear system. Hence  $(e_1, \dots, e_n)^\top = \mathcal{M}_p^{-1} \cdot \mathbf{t}_p$ . Finally, note that we only return this solution if all its coordinates are in  $\{0, 1\}^n$  and the total weighted sum is equal to the desired target, thus ensuring an incorrect solution is never returned. ◀

From now on we will focus on the proof of Theorem 2.

### 3.1 Generating Family

Let  $p \in \mathbb{N}$  be the prime fixed in Theorem 2. In particular,  $p$  is odd. Because the numbers  $a_1, \dots, a_n$  are generated at random in the interval  $\{1, \dots, \sqrt{\Gamma_{\text{LO}}}\}$  which has length much larger than  $p$ , the modular reductions  $a_i \pmod{p}$  are very close to uniform modulo  $p$ . For simplicity of the analysis we want to assume that they are uniform. This can be achieved by using the rejection sampling to discard any event when at least one  $a_i$  is greater than  $\sigma_p$ , where  $\sigma_p$  is the largest multiple of  $p$  such that  $\sigma_p < \sqrt{\Gamma_{\text{LO}}}$ . This is valid, because the probability that specific  $a_i$  is discarded is:

$$\mathbb{P}[a_i > \sigma_p] \leq \frac{|\sqrt{\Gamma_{\text{LO}}} - \sigma_p|}{\sqrt{\Gamma_{\text{LO}}}} \leq 2^{-\Omega(n \log n)}.$$

Therefore, by the union bound, probability that we do not discard any of  $a_1, \dots, a_n$  is at least  $1 - 2^{-\Omega(n \log n)}$ . Similarly, we can assume that  $a_1$  is not a multiple of  $p$ . Hence, there exists an integer  $a_1^{-1}$  such that  $a_1 \cdot a_1^{-1} \equiv_p 1$ . Note that this integer can be computed with  $\mathcal{O}(1)$  number of arithmetic operations.

Consider the lattice generated by the rows of the following matrix:

$$\mathcal{L} = \begin{pmatrix} a_1 & a_2 & a_3 & \cdots & a_n \\ p & 0 & 0 & \cdots & 0 \\ 0 & p & 0 & & 0 \\ 0 & 0 & p & & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & p \end{pmatrix} \in \mathbb{Z}^{(n+1) \times n} \quad (3)$$

Now, let us elaborate on the connection between matrix  $\mathcal{L}$  and any vector  $(\llbracket \mu a_1 \rrbracket_p, \dots, \llbracket \mu a_n \rrbracket_p)$  from the statement of Theorem 2.

► **Observation 5.** *For every integer  $\mu \in \mathbb{Z}$  it holds that:*

$$(\llbracket \mu a_1 \rrbracket_p, \dots, \llbracket \mu a_n \rrbracket_p) \in \mathcal{L}.$$

*Conversely, for every vector  $\mathbf{v} \in \mathcal{L} \cap \{-\lfloor p/2 \rfloor, \dots, \lfloor p/2 \rfloor\}^n$ , there exists  $\mu' \in \mathbb{Z}$  such that:*

$$\mathbf{v} = (\llbracket \mu' a_1 \rrbracket_p, \dots, \llbracket \mu' a_n \rrbracket_p).$$

*Moreover,  $\mu'$  is unique modulo  $p$  and its exact value can be determined with a constant number of arithmetic operations.*

**Proof.** For the first property observe that trivially  $\mu \cdot (a_1, \dots, a_n) \in \mathcal{L}$ . Let  $\mathbf{e}_i$  be the vector with 1 at the  $i$ th coordinate and 0 at the remaining coordinates. Note that by definition  $(p \cdot \mathbf{e}_i) \in \mathcal{L}$  for every  $i \in [n]$ . Then we construct the desired vector by

$$(\llbracket \mu a_1 \rrbracket_p, \dots, \llbracket \mu a_n \rrbracket_p) = \mu \cdot (a_1, \dots, a_n) - \sum_{i=1}^n \left\lfloor \frac{\mu a_i}{p} \right\rfloor \cdot (p \cdot \mathbf{e}_i) \in \mathcal{L}.$$

For the converse property, by definition any vector  $\mathbf{v} \in \mathcal{L}$  is represented as:

$$\mathbf{v} = k_0 \cdot (a_1, \dots, a_n) + \sum_{i=1}^n (k_i p) \cdot \mathbf{e}_i,$$

for some integers  $k_0, \dots, k_n \in \mathbb{Z}$ . Let us inspect the  $i$ th coordinate of  $\mathbf{v}$ . For every  $i \in [n]$  it holds that  $k_0 \cdot a_i + k_i \cdot p \in \{-\lfloor p/2 \rfloor, \dots, \lfloor p/2 \rfloor\}$ . Therefore, for every  $i \in [n]$ , the  $i$ th coordinate of  $\mathbf{v}$  is  $\llbracket k_0 a_i \rrbracket_p$ . We hence can set  $\mu' := k_0$ . Note, that given vector  $\mathbf{v}$ , the integer  $\mu'$  can be computed efficiently because it is expressed as  $\mu' = v_1 \cdot a_1^{-1}$ , where  $v_1$  is the first coordinate of  $\mathbf{v}$ . ◀

Now, let us further elaborate on the subsequent steps. We run the LLL algorithm on  $\mathcal{L}$ . The LLL algorithm returns a basis  $\mathbf{b}_1, \dots, \mathbf{b}_n$  of  $\mathcal{L}$ . We show that, with high probability,  $\|\mathbf{b}_i\|_1 \leq p/2$  for every  $i \in [n]$ . If that occurs we are nearly finished. By the Observation 5 each of the basis vectors  $\mathbf{b}_i$  is equal to  $(\llbracket \mu_i a_1 \rrbracket_p, \dots, \llbracket \mu_i a_n \rrbracket_p)$  for some integer  $\mu_i \in \mathbb{Z}$ . Moreover, because the vectors  $\mathbf{b}_1, \dots, \mathbf{b}_n$  form a basis of  $\mathcal{L}$  it means that the matrix formed by the vectors  $\mathbf{b}_1, \dots, \mathbf{b}_n$  has a full-rank.

Therefore, to complete the proof of Theorem 2 it suffices to show that  $\|\mathbf{b}_i\|_1 \leq p/2$  for every  $i \in [n]$ . By Cauchy–Schwarz inequality it is actually sufficient to prove that  $\|\mathbf{b}_i\|_2 \leq \frac{p}{2\sqrt{n}}$ .

### 3.2 Basis of Generating Family and Dual Basis

Lattice  $\mathcal{L}$  is given to us by a generating family in form of a rectangular matrix. It is much more convenient to work with a basis, especially when given by a square matrix. Therefore, we start by determining a basis of  $\mathcal{L}$ . As we have already noticed, we can assume without loss of generality that  $a_1$  is invertible in  $\mathbb{Z}_p$ . For every  $i \in 2, \dots, n$ , let  $\alpha_i = a_i a_1^{-1}$ . The rows of the following matrix form a basis of  $\mathcal{L}$ :

$$\mathcal{B}_0 = \begin{pmatrix} 1 & \alpha_2 & \alpha_3 & \cdots & \alpha_n \\ 0 & p & 0 & & 0 \\ 0 & 0 & p & & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & p \end{pmatrix} \in \mathbb{Z}^{n \times n}$$

In particular, this implies that the volume of  $\mathcal{L}$  is  $p^{n-1}$ . We can determine the dual lattice of  $\mathcal{L}$ , which in the full-dimensional case is spanned by the rows of the transpose of the inverse matrix. In our case, this is:

$$\mathcal{B}_0^\dagger = (\mathcal{B}_0^{-1})^T = \frac{1}{p} \cdot \begin{pmatrix} p & 0 & 0 & \cdots & 0 \\ -\alpha_2 & 1 & 0 & \cdots & 0 \\ -\alpha_3 & 0 & 1 & & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -\alpha_n & 0 & 0 & \dots & 1 \end{pmatrix} \in \mathbb{Q}^{n \times n}$$

Observe that  $p\mathcal{B}_0^\dagger$  generates the set of integral vectors  $(x_1, \dots, x_n) \in \mathbb{Z}^n$  such that  $\sum_{i=2}^n x_i \cdot \alpha_i \equiv_p -x_1$ . Or equivalently such that:

$$\sum_{i=1}^n x_i a_i \equiv_p 0.$$

### 3.3 Every vector of the LLL-reduced basis is probably short

Now, let  $\mathcal{B}$  be an LLL-reduced basis of the lattice  $\mathcal{L}$ . We denote its row vectors by  $\mathbf{b}_1, \dots, \mathbf{b}_n$  and their GSO by  $\mathbf{b}_1^*, \dots, \mathbf{b}_n^*$ . Recall that the dual basis, given by  $\mathcal{B}^\dagger = \mathbf{Rev}((\mathcal{B}^{-1})^T)$ .

In this section, we prove the following probabilistic property on a reduced basis:

## 57:12 Improving Lagarias-Odlyzko Algorithm

► **Lemma 6.** *With probability  $\geq 1 - 2^{-\Omega(n \log n)}$ , it holds that:*

$$\|\mathbf{b}_k^*\| \leq (\gamma_{\text{LLL}})^{\frac{n-1}{2}} \text{Vol}(\mathcal{L})^{1/n} \quad \text{for every } k \in \{1, \dots, n\}.$$

Lemma 6 can be proved as a consequence of the following inequalities, which hold with overwhelming probability when the integers  $a_1, \dots, a_n$  are drawn uniformly at random.

▷ **Claim 7.** *With probability  $\geq 1 - 2^{-\Omega(n \log n)}$  it holds that:*

$$\|\mathbf{b}_n^*\| \leq \text{Vol}(\mathcal{L})^{1/n} \leq \|\mathbf{b}_1^*\|.$$

**Proof of Lemma 6 assuming Claim 7.** Let us assume that both inequalities in Claim 7 hold. If  $k \geq \frac{n+1}{2}$  then by repeated application of Inequality (1) we have that  $\|\mathbf{b}_k^*\| \leq (\gamma_{\text{LLL}})^{n-k} \|\mathbf{b}_n^*\| \leq (\gamma_{\text{LLL}})^{\frac{n-1}{2}} \text{Vol}(\mathcal{L})^{1/n}$ , which is the desired result. Hence we need to focus on the case where  $k < \frac{n+1}{2}$ . Observe that by repeated application of Inequality (1) and Claim 7 we have

$$\|\mathbf{b}_1^*\| \cdots \|\mathbf{b}_{k-1}^*\| \geq \prod_{i=1}^{k-1} \frac{\|\mathbf{b}_i^*\|}{(\gamma_{\text{LLL}})^{i-1}} = (\gamma_{\text{LLL}})^{-\frac{(k-1)(k-2)}{2}} \cdot \|\mathbf{b}_1^*\|^{k-1} \geq (\gamma_{\text{LLL}})^{-\frac{(k-1)(k-2)}{2}} \cdot \text{Vol}(\mathcal{L})^{\frac{k-1}{n}}.$$

Similarly, we have

$$\|\mathbf{b}_k^*\| \cdots \|\mathbf{b}_n^*\| \geq \prod_{i=0}^{n-k} \frac{\|\mathbf{b}_i^*\|}{(\gamma_{\text{LLL}})^i} = (\gamma_{\text{LLL}})^{-\frac{(n-k)(n-k+1)}{2}} \cdot \|\mathbf{b}_k^*\|^{n-k+1}.$$

Recall that the volume of the lattice  $\mathcal{L}$  is  $\text{Vol}(\mathcal{L}) = \|\mathbf{b}_1^*\| \cdots \|\mathbf{b}_n^*\|$ . Hence by multiplying the above inequalities we have

$$\text{Vol}(\mathcal{L}) \geq \text{Vol}(\mathcal{L})^{\frac{k-1}{n}} \cdot \|\mathbf{b}_k^*\|^{n-k+1} \cdot (\gamma_{\text{LLL}})^{-\frac{(n-k)(n-k+1)}{2} - \frac{(k-1)(k-2)}{2}}.$$

Because  $k < \frac{n+1}{2}$  we have that  $(n-k)(n-k+1) + (k-1)(k-2) \leq (n-1)(n-k+1)$ . Hence

$$\text{Vol}(\mathcal{L}) \geq \text{Vol}(\mathcal{L})^{\frac{k-1}{n}} \cdot \|\mathbf{b}_k^*\|^{n-k+1} \cdot (\gamma_{\text{LLL}})^{-\frac{(n-1)(n-k+1)}{2}}.$$

By rearranging the terms we have

$$\text{Vol}(\mathcal{L})^{\frac{n-k+1}{n}} \cdot (\gamma_{\text{LLL}})^{\frac{(n-1)(n-k+1)}{2}} \geq \|\mathbf{b}_k^*\|^{n-k+1}.$$

After taking  $(n-k+1)$ th root, this yields the desired inequality. ◀

It remains to prove inequalities in Claim 7. We split this proof into Proposition 8 and Proposition 9. First, we focus on the right side of the inequality.

► **Proposition 8.**

$$\Pr \left[ \|\mathbf{b}_1^*\| < \text{Vol}(\mathcal{L})^{1/n} \right] < 2^{-\Omega(n \log n)}.$$

**Proof.** Recall that  $\text{Vol}(\mathcal{L}) = p^{n-1}$  and  $\mathbf{b}_1 = \mathbf{b}_1^*$ . Let us bound the probability of  $\|\mathbf{b}_1\| < p^{\frac{n-1}{n}}$ . Note that in this case, Observation 5 asserts that there exists  $\mu \in \mathbb{Z}$  such that

$$\|(\llbracket \mu a_1 \rrbracket_p, \dots, \llbracket \mu a_n \rrbracket_p)\| < p^{\frac{n-1}{n}}.$$

Moreover, integer  $\mu$  is determined exactly as  $\mu := (\mathbf{b}_1)_1 \cdot a_1^{-1}$ . Hence, for a fixed  $\mathbf{y} \in \{-\lfloor p/2 \rfloor, \dots, \lfloor p/2 \rfloor\}^n$  the numbers  $a_2, \dots, a_n$  via  $\llbracket \cdot \rrbracket_p$  operations are determined. This means that for a given value  $a_1$ , the conditional probability for a fixed  $\mathbf{y} \in \{-\lfloor p/2 \rfloor, \dots, \lfloor p/2 \rfloor\}^n$  is:

$$\Pr [\mathbf{y} = (\llbracket \mu a_1 \rrbracket_p, \dots, \llbracket \mu a_n \rrbracket_p)] = \frac{1}{p^{n-1}},$$

which does not depend on  $a_1$ . Thus the unconditional probability is  $1/p^{n-1}$  as well. On the other hand, the number of vectors  $\mathbf{y}$  such that  $\|\mathbf{y}\| < p^{\frac{n-1}{n}}$  is  $\leq \text{Vol}(\text{Ball}_n(p^{\frac{n-1}{n}} + \sqrt{n}))$ . Therefore by the union bound we conclude:

$$\Pr \left[ \|\mathbf{b}_1\| \leq p^{\frac{n-1}{n}} \right] < \frac{\text{Vol}(\text{Ball}_n(p^{\frac{n-1}{n}} + \sqrt{n}))}{p^{n-1}} < 2^{-\Omega(n \log n)}. \quad \blacktriangleleft$$

For the other part of the inequality in Claim 7, we need to prove that:

► **Proposition 9.**

$$\Pr \left[ \|\mathbf{b}_n^*\| > \text{Vol}(\mathcal{L})^{1/n} \right] < 2^{-\Omega(n \log n)}.$$

**Proof.** Again recall that  $\text{Vol}(\mathcal{L}) = p^{n-1}$  and our goal is to bound the probability that  $\|\mathbf{b}_n^*\| > p^{1-1/n}$ . Now, we inspect the dual lattice  $\mathcal{L}^\dagger$  and the dual basis  $\mathcal{B}^\dagger$  of the reduced basis. Recall that  $(\mathbf{b}_1^\dagger)^* = \mathbf{b}_1^\dagger$  and  $\|\mathbf{b}_n^*\| = \|\mathbf{b}_1^\dagger\|^{-1}$ . Therefore, we aim to bound the probability  $\|\mathbf{b}_1^\dagger\| < \frac{p^{1/n}}{p}$ .

Because  $\mathbf{b}_1^\dagger \in \mathcal{L}^\dagger$ , there exists  $\mathbf{s} = (s_1, \dots, s_n) \in \mathbb{Z}^n$  such that:

$$\mathbf{b}_1^\dagger = \frac{1}{p} \cdot \left( \left( p \cdot s_1 - \sum_{i=2}^n \alpha_i \cdot s_i \right), s_2, s_3, \dots, s_n \right).$$

First, remark that when the length of  $\mathbf{b}_1^\dagger$  is bounded by  $\frac{p^{1/n}}{p}$  it holds that:

$$\|(s_2, \dots, s_n)\| < p^{1/n}.$$

Observe that the number of  $(s_2, \dots, s_n) \in \mathbb{Z}^{n-1}$  of length  $< p^{1/n}$  is at most  $\text{Vol}(\text{Ball}_{n-1}(p^{1/n} + \sqrt{n}))$ . Hence, from now on, we fix coordinates  $s_2, \dots, s_n$  and examine the probability that the first coordinate of  $\mathbf{b}_1^\dagger$  is bounded. In particular, when  $\|\mathbf{b}_1^\dagger\| < \frac{p^{1/n}}{p}$  it holds that:

$$\left| p s_1 - \sum_{i=2}^n \alpha_i s_i \right| < p^{1/n}$$

Note that, if the values  $s_2, \dots, s_n$  are fixed then  $s_1$  is determined. Hence, it holds that:

$$\sum_{i=2}^n \alpha_i s_i \equiv k \pmod{p} \text{ for some } k \in [p^{1/n}]. \quad (4)$$

Recall that the numbers  $a_1, \dots, a_n$  are selected uniformly at random from a range greater than  $[p]$ . Hence, for a fixed  $\mathbf{s}$  the probability that Equation 4 holds is  $\frac{p^{1/n}}{p}$ . Therefore, by union bound we have that

$$\Pr \left[ \|\mathbf{b}_1^\dagger\| < \frac{p^{1/n}}{p} \right] \leq \frac{p^{1/n}}{p} \cdot \text{Vol}(\text{Ball}_{n-1}(p^{1/n} + \sqrt{n})) < 2^{-\Omega(n \log n)}. \quad \blacktriangleleft$$

Applying the union bounds on the two propositions gives an upper bound on the negation of the event considered in Claim 7. This concludes the proof of Claim 7.

### 3.4 Proof of Theorem 2

**Proof.** By Lemma 6, with probability  $\geq 1 - 2^{-\Omega(n \log n)}$ , for every  $i \in [n]$  we have:

$$\|\mathbf{b}_i^*\| \leq (\gamma_{\text{LLL}})^{\frac{n-1}{2}} p^{\frac{n-1}{n}}.$$

We set  $p = \Theta((\gamma_{\text{LLL}})^{n^2/2} \cdot 2^{2n \log_2 n})$ . This means that for every  $i \in [n]$  it holds that:

$$\|\mathbf{b}_i^*\| \leq \frac{p}{n^2}.$$

The vectors  $\mathbf{b}_1^*, \dots, \mathbf{b}_n^*$  are the GSO basis of  $\mathbf{b}_1, \dots, \mathbf{b}_n \in \mathbb{Z}^n$  and do not necessarily have integral components. We want to bound the lengths of the  $\mathbf{b}_1, \dots, \mathbf{b}_n$  in the LLL-reduced basis. By definition of GSO basis, we have

$$\mathbf{b}_i = \mathbf{b}_i^* + \sum_{j=1}^{i-1} \mu_{i,j} \mathbf{b}_j^*.$$

The size reduction condition of LLL guarantees that  $|\mu_{i,j}| \leq \mu < 1$ . Therefore, for every  $i \in [n]$  it holds that:

$$\|\mathbf{b}_i\| \leq n \cdot \max_{k \in [n]} \|\mathbf{b}_k^*\| < \frac{p}{n}.$$

By the Cauchy-Schwartz inequality, the fact that  $\|\mathbf{b}_i\| < p/n$  implies that  $\|\mathbf{b}_i\|_1 < p/\sqrt{n}$ . In particular, it means that  $\mathbf{b}_i \in \{-\lfloor p/2 \rfloor, \dots, \lfloor p/2 \rfloor\}^n$ . Therefore, by Observation 5 we can find integers  $\mu_1, \dots, \mu_n$  such that:

$$\mathbf{b}_i = (\lfloor \mu_i a_1 \rfloor_p, \dots, \lfloor \mu_i a_n \rfloor_p).$$

for every  $i \in [n]$ . Note that these vectors are linearly independent because the vectors  $\mathbf{b}_i$  form a basis.

For the running time, observe that we need a single call to the LLL algorithm to get the basis  $\mathbf{b}_1, \dots, \mathbf{b}_n$  and a linear number of arithmetic operations to retrieve the coefficients  $\mu_1, \dots, \mu_n$ . This concludes the proof of Theorem 2.  $\blacktriangleleft$

---

#### References

- 1 Divesh Aggarwal, Antoine Joux, Miklos Santha, and Karol Węgrzycki. Polynomial time algorithms for integer programming and unbounded subset sum in the total regime, 2024. doi:10.48550/arXiv.2407.05435.
- 2 Divesh Aggarwal, Zeyong Li, and Noah Stephens-Davidowitz. A  $2^{n/2}$ -Time Algorithm for  $\sqrt{n}$ -SVP and  $\sqrt{n}$ -Hermite SVP, and an Improved Time-Approximation Tradeoff for (H)SVP. In *Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, volume 12696 of *Lecture Notes in Computer Science*, pages 467–497. Springer, 2021. doi:10.1007/978-3-030-77870-5\_17.
- 3 Jonathan Allcock, Yassine Hamoudi, Antoine Joux, Felix Klingelhöfer, and Miklos Santha. Classical and quantum algorithms for variants of subset-sum via dynamic programming. In *30th Annual European Symposium on Algorithms, ESA 2022*, volume 244 of *LIPICs*, pages 6:1–6:18. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPICs.ESA.2022.6.
- 4 Per Austrin, Petteri Kaski, Mikko Koivisto, and Jussi Määttä. Space-Time Tradeoffs for Subset Sum: An Improved Worst Case Algorithm. In *Automata, Languages, and Programming - 40th International Colloquium, ICALP 2013*, pages 45–56, 2013. doi:10.1007/978-3-642-39206-1\_5.

- 5 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*, pages 48–61, 2015. doi:10.4230/LIPIcs.STACS.2015.48.
- 6 Per Austrin, Petteri Kaski, Mikko Koivisto, and Jesper Nederlof. Dense Subset Sum May Be the Hardest. In *33rd Symposium on Theoretical Aspects of Computer Science, STACS 2016*, pages 13:1–13:14, 2016. doi:10.4230/LIPIcs.STACS.2016.13.
- 7 Eleonore Bach, Friedrich Eisenbrand, Thomas Rothvoss, and Robert Weismantel. Forall-exist statements in pseudopolynomial time. In *Proceedings of the 2025 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2225–2233. SIAM, 2025. doi:10.1137/1.9781611978322.73.
- 8 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. doi:10.1137/17M1158203.
- 9 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. doi:10.1007/978-3-642-20465-4\_21.
- 10 Anja Becker, Antoine Joux, Alexander May, and Alexander Meurer. Decoding random binary linear codes in  $2^{n/20}$ : How  $1+1=0$  improves information set decoding. In *Advances in Cryptology–EUROCRYPT 2012: 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15–19, 2012. Proceedings 31*, pages 520–536. Springer, 2012. doi:10.1007/978-3-642-29011-4\_31.
- 11 Richard Bellman. *Dynamic Programming*. Princeton University Press, Princeton, NJ, USA, 1957.
- 12 Daniel J Bernstein, Stacey Jeffery, Tanja Lange, and Alexander Meurer. Quantum algorithms for the subset-sum problem. In *Post-Quantum Cryptography: 5th International Workshop, PQCrypto 2013, Limoges, France, June 4–7, 2013. Proceedings 5*, pages 16–33. Springer, 2013. doi:10.1007/978-3-642-38616-9\_2.
- 13 Elena Böhme. *Verbesserte subset-sum algorithmen*. PhD thesis, Master’s thesis, Ruhr Universität Bochum, 2011.
- 14 Xavier Bonnetain, Rémi Bricout, André Schrottenloher, and Yixin Shen. Improved classical and quantum algorithms for subset-sum. In *Advances in Cryptology–ASIACRYPT 2020: 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7–11, 2020, Proceedings, Part II 26*, pages 633–666. Springer, 2020. doi:10.1007/978-3-030-64834-3\_22.
- 15 Murray R Bremner. *Lattice basis reduction: an introduction to the LLL algorithm and its applications*. CRC Press, 2011.
- 16 Ernest F Brickell. Solving low density knapsacks. In *Advances in Cryptology: Proceedings of Crypto 83*, pages 25–37. Springer, 1984.
- 17 Karl Bringmann. A Near-Linear Pseudopolynomial Time Algorithm for Subset Sum. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1073–1084. SIAM, 2017. doi:10.1137/1.9781611974782.69.
- 18 Karl Bringmann. Knapsack with small items in near-quadratic time. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 259–270, 2024. doi:10.1145/3618260.3649719.
- 19 Lin Chen, Yuchen Mao, and Guochuan Zhang. Long arithmetic progressions in sumsets and subset sums: Constructive proofs and efficient witnesses. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, pages 2086–2097, 2025. doi:10.1145/3717823.3718281.
- 20 Xi Chen, Yaonan Jin, Tim Randolph, and Rocco A. Servedio. Average-case subset balancing problems. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 743–778, 2022. doi:10.1137/1.9781611977073.33.

- 21 Don Coppersmith. Finding a small root of a univariate modular equation. In *Advances in Cryptology - EUROCRYPT '96, International Conference on the Theory and Application of Cryptographic Techniques*, volume 1070 of *Lecture Notes in Computer Science*, pages 155–165. Springer, 1996. doi:10.1007/3-540-68339-9\_14.
- 22 Matthijs J Coster, Antoine Joux, Brian A LaMacchia, Andrew M Odlyzko, Claus-Peter Schnorr, and Jacques Stern. Improved low-density subset sum algorithms. *Computational complexity*, 2:111–128, 1992. doi:10.1007/BF01201999.
- 23 Itai Dinur, Orr Dunkelman, Nathan Keller, and Adi Shamir. Efficient Dissection of Composite Problems, with Applications to Cryptanalysis, Knapsacks, and Combinatorial Search Problems. In *Advances in Cryptology - CRYPTO 2012 - 32nd Annual Cryptology Conference. Proceedings*, 2012.
- 24 Alan M. Frieze. On the Lagarias-Odlyzko Algorithm for the Subset Sum Problem. *SIAM J. Comput.*, 15(2):536–539, 1986. doi:10.1137/0215038.
- 25 Merrick L Furst and Ravi Kannan. Succinct certificates for almost all subset sum problems. *SIAM Journal on Computing*, 18(3):550–558, 1989. doi:10.1137/0218037.
- 26 Steven D Galbraith. *Mathematics of public key cryptography*. Cambridge University Press, 2012.
- 27 Nicolas Gama and Phong Q Nguyen. Finding short lattice vectors within Mordell’s inequality. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 207–216, 2008. doi:10.1145/1374376.1374408.
- 28 Alexander Helm and Alexander May. Subset sum quantumly in  $1.17^n$ . In *13th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2018)*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPIcs.TQC.2018.5.
- 29 Ellis Horowitz and Sartaj Sahni. Computing Partitions with Applications to the Knapsack Problem. *J. ACM*, 21(2):277–292, 1974. doi:10.1145/321812.321823.
- 30 Nick Howgrave-Graham. Finding small roots of univariate modular equations revisited. In *Cryptography and Coding, 6th IMA International Conference*, volume 1355, pages 131–142. Springer, 1997. doi:10.1007/BFB0024458.
- 31 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. doi:10.1007/978-3-642-13190-5\_12.
- 32 Ce Jin. 0-1 Knapsack in nearly quadratic time. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 271–282, 2024. doi:10.1145/3618260.3649618.
- 33 Antoine Joux, Hunter Kippen, and Julian Loss. A concrete analysis of wagner’s k-list algorithm over  $\mathbb{Z}_p$ , 2024.
- 34 Antoine Joux and Jacques Stern. Improving the Critical Density of the Lagarias-Odlyzko Attack Against Subset Sum Problems. In *Fundamentals of Computation Theory, 8th International Symposium, FCT '91*, volume 529, pages 258–264. Springer, 1991. doi:10.1007/3-540-54458-5\_70.
- 35 Antoine Joux and Karol Wegrzycki. Improving lagarias-odlyzko algorithm for average-case subset sum: Modular arithmetic approach. *CoRR*, abs/2408.16108, 2024. doi:10.48550/arXiv.2408.16108.
- 36 Hans Kellerer, Ulrich Pferschy, and David Pisinger. *Knapsack problems*. Springer, 2004. doi:10.1007/978-3-540-24777-7.
- 37 Konstantinos Koiliaris and Chao Xu. Faster pseudopolynomial time algorithms for subset sum. *ACM Transactions on Algorithms (TALG)*, 15(3):1–20, 2019. doi:10.1145/3329863.
- 38 J. C. Lagarias and Andrew M. Odlyzko. Solving low-density subset sum problems. *J. ACM*, 32(1):229–246, 1985. doi:10.1145/2455.2461.
- 39 Brian A LaMacchia. Basis reduction algorithms and subset sum problems. Technical report, Massachusetts Institute of Technology, 1991.
- 40 Arjen K Lenstra, Hendrik Willem Lenstra, and László Lovász. Factoring polynomials with rational coefficients. *Mathematische annalen*, 261:515–534, 1982.

- 41 Ralph C. Merkle and Martin E. Hellman. Hiding information and signatures in trapdoor knapsacks. *IEEE Trans. Information Theory*, 24(5):525–530, 1978. doi:10.1109/TIT.1978.1055927.
- 42 Jesper Nederlof and Karol Węgrzycki. Improving Schroeppe and Shamir’s Algorithm for Subset Sum via Orthogonal Vectors. In *STOC ’21: 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1670–1683. ACM, 2021. doi:10.1145/3406325.3451024.
- 43 Phong Q Nguyen and Brigitte Vallée. *The LLL algorithm*. Springer, 2010.
- 44 Andrew M Odlyzko. The rise and fall of knapsack cryptosystems. *Cryptology and computational number theory*, 42(2), 1990.
- 45 Adam Polak, Lars Rohwedder, and Karol Węgrzycki. Knapsack and Subset Sum with Small Items. In *48th International Colloquium on Automata, Languages, and Programming, ICALP 2021*, volume 198 of *LIPIcs*, pages 106:1–106:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.ICALP.2021.106.
- 46 Stanislaw Radziszowski and Donald Kreher. Solving subset sum problems with the  $L^3$  algorithm. *The Charles Babbage Research Centre: The Journal of Combinatorial Mathematics and Combinatorial Computing*, 3, 1988.
- 47 Keegan Ryan and Nadia Heninger. Fast practical lattice reduction through iterated compression. In *Advances in Cryptology – CRYPTO 2023*, pages 3–36, Cham, 2023. Springer Nature Switzerland. doi:10.1007/978-3-031-38548-3\_1.
- 48 Claus-Peter Schnorr. A hierarchy of polynomial time lattice basis reduction algorithms. *Theoretical computer science*, 53(2-3):201–224, 1987. doi:10.1016/0304-3975(87)90064-8.
- 49 Claus-Peter Schnorr and Martin Euchner. Lattice basis reduction: Improved practical algorithms and solving subset sum problems. *Mathematical programming*, 66:181–199, 1994. doi:10.1007/BF01581144.
- 50 Richard Schroeppe and Adi Shamir. A  $T=O(2^{n/2})$ ,  $S=O(2^{n/4})$  Algorithm for Certain NP-Complete Problems. *SIAM J. Comput.*, 10(3):456–464, 1981.
- 51 Andrew Shallue. An improved multi-set algorithm for the dense subset sum problem. In *International Algorithmic Number Theory Symposium*, pages 416–429. Springer, 2008. doi:10.1007/978-3-540-79456-1\_28.

## A Extension to the block reduction

In this Section, we prove Theorem 3. Note that the proof of Theorem 1 and the proof of Theorem 2 only relied on the specific lattice reduction via Lemma 6. Hence, in order to establish Theorem 3, it is enough to generalize Lemma 6 to the case of block basis reduction and prove the following statement.

► **Lemma 10.** *Let  $\mathbf{b}_1, \dots, \mathbf{b}_n$  be the reduced basis obtained by running the block lattice reduction algorithm of [27] with parameter  $\Upsilon$  on the lattice  $\mathcal{L}$  defined in (3). With probability  $\geq 1 - 2^{-\Omega(n \log n)}$ , it holds that:*

$$\|\mathbf{b}_k^*\| \leq C_\Upsilon \cdot \Upsilon^{\frac{n-1}{2}} \cdot \text{Vol}(\mathcal{L})^{1/n} \quad \text{for every } k \in \{1, \dots, n\},$$

where  $C_\Upsilon$  is a constant that only depends on  $\Upsilon$ .

Before we prove Lemma 10, let us elaborate on the block basis reduction from [27]. Here, we follow the description from the textbook [43, Chapter 2, Sliding Algorithm]. The main parameter in the block-reduction is the block size  $w > 2$ . Gama and Nguyen [27], actually parameterize their reduction with respect to the block-size. For any integer  $k$  let  $\gamma_k$  be the Hermite constant (see Chapter 2 [43]). The blocksize  $w$  of the lattice reduction of Gama and Nguyen is the smallest integer such that  $\Upsilon \geq (\gamma_w)^{1/(w-1)}$ . Note that  $\gamma_n = \Theta(n)$ , so  $w$  is a properly defined constant that depends on the choice of  $\Upsilon$ .

## 57:18 Improving Lagarias-Odlyzko Algorithm

The block-reduction of [27] has two important properties. First, it returns a basis that is *block-Mordell-reduced*. The only property about block-Mordell-reduced basis we need is the following inequality:

▷ **Claim 11** (Primal-Dual inequality, Chapter 2, Lemma 11, Equality (2.48) in [43]). Let  $\mathbf{b}_1, \dots, \mathbf{b}_n$  be a block-Mordell-reduced basis of the lattice  $L$  with blocksize  $w \geq 2$ , then:

$$\frac{\|\mathbf{b}_j^*\|}{\|\mathbf{b}_{j+w}^*\|} \leq (\gamma_w)^{w/(w-1)} \leq \Upsilon^w \quad (5)$$

where  $j \in \{1, \dots, n-w\}$  such that  $j \equiv_w 1$ .

The second property of the algorithm from [27] is that its output basis is also LLL-reduced (see [43, Chapter 2, Sliding Algorithm, Algorithm 6]). As a consequence, we can also use inequality (1).

▷ **Claim 12.** For all  $1 \leq j < i \leq n$ , it holds that

$$\|\mathbf{b}_i^*\| \geq (\gamma_{\text{LLL}})^{-2w} \cdot \Upsilon^{j-i} \cdot \|\mathbf{b}_j^*\|. \quad (6)$$

*Proof.* Note that blocks in block-reduced algorithm overlap on the indices  $\equiv_w 1$  [27]. Hence, for the sake of clarity for  $i \in [n]$  let  $q(i) \in \mathbb{N}$  to be the index of the block that  $i$  is contained, i.e.,  $i \in [q(i)w + 1, (q(i) + 1)w]$ . Let  $\ell(i) = q(i)w + 1$  be the index of the first element in the  $i$ th block. Similarly let  $r(i) = \ell(i)$  when  $i \equiv_w 1$  and  $r(i) = \min\{n, \ell(i) + 1\}$  otherwise, be the index of the last element of the block.

Because  $\mathbf{b}_1, \dots, \mathbf{b}_n$  is also LLL reduced we know that that by repeated application of inequality (1), for every  $k \in [n]$  it holds:

$$(\gamma_{\text{LLL}})^{-w} \|\mathbf{b}_{\ell(k)}^*\| \leq \|\mathbf{b}_k^*\| \leq (\gamma_{\text{LLL}})^w \|\mathbf{b}_{r(k)}^*\|. \quad (7)$$

Next, we use Equation (5), that for  $j < i$  says:

$$\|\mathbf{b}_{\ell(i)}^*\| \geq \Upsilon^{(\ell(i)-r(j))w} \cdot \|\mathbf{b}_{r(j)}^*\|. \quad (8)$$

Hence, by combining above we get:

$$\begin{aligned} \|\mathbf{b}_i^*\| &\geq (\gamma_{\text{LLL}})^{-w} \cdot \|\mathbf{b}_{\ell(i)}^*\| && \text{(by (7))} \\ &\geq (\gamma_{\text{LLL}})^{-w} \cdot \Upsilon^{(\ell(i)-r(j))w} \cdot \|\mathbf{b}_{r(j)}^*\| && \text{(by (8))} \\ &\geq (\gamma_{\text{LLL}})^{-2w} \cdot \Upsilon^{i-j} \cdot \|\mathbf{b}_j^*\|, && \text{(by (7))} \end{aligned}$$

where the last inequality follows because  $\ell(i)w - r(j)w \geq i - j$ . ◁

**Proof of Lemma 10.** The probabilistic event in Claim 7 is a function of the lattice  $\mathcal{L}$  itself and not of any specific basis of it. Thus, it remains valid for the basis produced by the block reduction algorithm. Hence, with  $\geq 1 - 2^{-\Omega(n \log n)}$  we have that  $\|\mathbf{b}_n^*\| \leq \text{Vol}(\mathcal{L})^{1/n} \leq \|\mathbf{b}_1^*\|$ . First, consider the case  $k \geq \frac{n+1}{2}$ .

$$\|\mathbf{b}_k^*\| \leq (\gamma_{\text{LLL}})^{2w} \cdot \Upsilon^{n-k} \|\mathbf{b}_n^*\|.$$

This concludes the proof for the case  $k \geq \frac{n+1}{2}$  as  $n - k \leq (n - 1)/2$ . It remains to consider the case  $k < \frac{n+1}{2}$ . By repeated application of (6) we have:

$$\begin{aligned} \|\mathbf{b}_1^*\| \cdots \|\mathbf{b}_{k-1}^*\| &\geq \prod_{i=1}^{k-1} (\gamma_{\text{LLL}})^{-2w} \cdot \Upsilon^{-(i-1)} \|\mathbf{b}_1^*\| \\ &\geq (\gamma_{\text{LLL}})^{-2w(k-1)} \cdot \Upsilon^{-(k-2)(k-1)/2} \cdot \|\mathbf{b}_1^*\|^{k-1}. \end{aligned}$$

Similarly:

$$\begin{aligned} \|\mathbf{b}_k^*\| \cdots \|\mathbf{b}_n^*\| &\geq \prod_{i=k}^n (\gamma_{\text{LLL}})^{-2w} \cdot \Upsilon^{-(i-k)} \|\mathbf{b}_k^*\| \\ &\geq (\gamma_{\text{LLL}})^{-2w(n-k)} \cdot \Upsilon^{-(n-k)(n-k+1)/2} \cdot \|\mathbf{b}_k^*\|^{n-k}. \end{aligned}$$

Hence, by multiplying these two inequalities and by  $\prod_{i=1}^n \|\mathbf{b}_i^*\| = \text{Vol}(\mathcal{L})$  we conclude that:

$$\text{Vol}(\mathcal{L}) \geq (\gamma_{\text{LLL}})^{-2w(n-1)} \cdot \Upsilon^{-(n-k)(n-k+1)/2 - (k-2)(k-1)/2} \cdot \|\mathbf{b}_k^*\|^{n-k+1} \cdot \|\mathbf{b}_1^*\|^{k-1}.$$

Because  $k < \frac{n+1}{2}$  we have that  $(n-k)(n-k+1) + (k-1)(k-2) \leq (n-1)(n-k+1)$ . Hence,

$$\text{Vol}(\mathcal{L}) \geq (\gamma_{\text{LLL}})^{-2w(n-1)} \cdot \Upsilon^{-(n-1)(n-k+1)/2} \cdot \|\mathbf{b}_k^*\|^{n-k+1} \|\mathbf{b}_1^*\|^{k-1}.$$

Next, we use Claim 7 with  $\|\mathbf{b}_1^*\| \geq \text{Vol}(\mathcal{L})^{1/n}$  to have

$$\text{Vol}(\mathcal{L})^{\frac{n-k+1}{n}} \geq (\gamma_{\text{LLL}})^{-2w(n-1)} \cdot \Upsilon^{-((n-1)(n-k+1)/2)} \cdot \|\mathbf{b}_k^*\|^{n-k+1}.$$

After taking  $(n-k+1)$ th root we conclude that:

$$(\gamma_{\text{LLL}})^{2w(n-1)/(n-k+1)} \cdot \Upsilon^{(n-1)/2} \cdot \text{Vol}(\mathcal{L}) \geq \|\mathbf{b}_k^*\|.$$

Finally, observe that as  $k < (n+1)/2$  this means that  $(\gamma_{\text{LLL}})^{2w(n-1)/(n-k+1)} \leq (\gamma_{\text{LLL}})^{4w} = C_\Upsilon$  is a constant that depends only on  $\Upsilon$ , since  $\gamma_{\text{LLL}}$  is a constant and  $w$  depends only on  $\Upsilon$ . Hence:

$$C_\Upsilon \cdot \Upsilon^{(n-1)/2} \cdot \text{Vol}(\mathcal{L})^{1/n} \geq \|\mathbf{b}_k^*\|. \quad \blacktriangleleft$$