Verification of NP-Hardness Reduction Functions for Exact Lattice Problems

Katharina Kreuzer(✉) and Tobias Nipkow

Technical University of Munich, Boltzmannstr. 3, 85748 Garching, Germany
k.kreuzer@tum.de

Abstract. This paper describes the formal verification of NP-hardness reduction functions of two key problems relevant in algebraic lattice theory: the closest vector problem and the shortest vector problem, both in the infinity norm. The formalization uncovered a number of problems with the existing proofs in the literature. The paper describes how these problems were corrected in the formalization. The work was carried out in the proof assistant Isabelle.

Keywords: verification · NP-hardness · lattice problems · integer programming

1 Introduction

In recent years, algebraic lattices have received increasing attention for their use in post-quantum cryptography. Algebraic lattices are additive, discrete subgroups of $\mathbb{R}^n$, i.e. a set of points in $\mathbb{R}^n$ with certain structures. One can also define lattices over finite fields, rings or modules as used in many modern post-quantum crypto systems such as the CRYSTALS suites, NTRU and Saber.

Two problems form the very basis for computationally hard problems on lattices, namely the closest vector problem (CVP) and the shortest vector problem (SVP). Given a finite set of basis vectors in $\mathbb{R}^n$, the set of all linear combinations with integer coefficients forms a lattice. In optimization form, the SVP asks for the shortest vector in the lattice and the CVP asks for the lattice vector closest to some given target vector, both with respect to some given norm.

When working over the reals, the p-norm (for $p \geq 1$) is defined as $\sqrt[p]{\sum_i |x_i|^p}$. The most common examples are the Euclidean norm $\|x\|_2$ and the infinity norm $\|x\|_\infty = \max_i\{|x_i|\}$, which is the limit for $p \to \infty$.

We have formalized, corrected and verified a number of NP-hardness proofs from the literature, uncovering a number of mistakes along the way. The first NP-hardness proof of the CVP and SVP in infinity norm is due to van Emde-Boas [7]. For other norms (especially for the Euclidean norm), there is only a randomized reduction for the NP-hardness of the SVP so far [2]. For the CVP,

This work was supported by the Research Training Group GRK 2428 CONVEY of the German Research Council (DFG).

B. Pientka and C. Tinelli (Eds.): CADE 2023, LNAI 14132, pp. 365–381, 2023.
https://doi.org/10.1007/978-3-031-38499-8_21

NP-hardness has been shown in any p-norm for $p \geq 1$. One exemplary proof can be found in the book by Micciancio and Goldwasser [15, Chapter 3, Thm 3.1].

The CVP and SVP were the starting point for lattice-based post-quantum cryptography [16]. Moreover, the relevance of these problems can also be seen from the rich literature on approximation results. For example, the LLL-algorithm by Lenstra, Lenstra and Lovász [12] gives a polynomial-time algorithm for lattice basis reduction which solves integer linear programs in fixed dimensions. Using this reduced basis, one can find good approximations to the CVP using Babai's algorithm [3] for certain approximation factors. Still, for arbitrary dimensions, the problem remains NP-hard. Further approximation results for the CVP, SVP and integer programming can be found elsewhere [6,9,10,14,19]. These approximation problems are used in cryptography. However, we will focus on the exact CVP and SVP in this paper.

A number of more basic NP-hardness proofs have been formalized in several theorem provers so far. For example, there are formalizations of the Cook-Levin Theorem in Coq [8] and Isabelle [4]. Formalizing Karp's 21 NP-hard problems (including the Subset Sum and Partition Problems assumed to be NP-hard in this paper) in Isabelle is an ongoing project.

1.1 Contributions

In this paper we present NP-hardness proofs of the CVP and SVP in infinity norm that have been verified in a proof assistant. We roughly follow the book by Micciancio and Golwasser [15, Chapter 3, Thm 3.1] and the report by van Emde-Boas [7]. However, many problems with the original proofs were encountered during the formalization efforts. We will have a look at different approaches and their advantages or problems.

We also verified the proof of NP-hardness of the CVP for any finite $p \geq 1$ from the book by Micciancio and Goldwasser. This verification did not uncover any problems with the informal proof. Thus we do not discuss it in detail.

These formalizations were carried out with the help of the proof assistant Isabelle [17,18] and are available online [11]. They comprise 5200 lines. To the authors knowledge, they are the first formalizations of hardness proofs for lattice problems. Because of the importance of the SVP and CVP and the problems in existing proofs, we consider our proofs a contribution to the foundations of verified cryptography. However, we do not claim that these hardness results directly imply quantum-resistance of any lattice-based cryptosystems.

1.2 Overview

The paper is structured as follows. Section 2 introduces the foundations. The rest of the paper is dedicated to the proofs, which are phrased as the following two polynomial time reduction chains:

- Subset Sum $\leq_p$ CVP
- Partition $\leq_p$ Bounded Homogeneous Linear Equations $\leq_p$ SVP

Subset Sum and Partition are famous fundamental problems whose NP-hardness has been proved many times in the literature and which we take for granted.

Section 3 presents the reduction of Subset Sum to the CVP. Differences between our formalization and the book by Micciancio and Goldwasser [15] are presented with examples that demonstrate problems with the original proof. Moreover, an example is given why the generalization to the SVP given in [15] does not work.

Therefore we turn to the early proof of NP-hardness of the SVP by van Emde Boas [7]. This proof uses the Bounded Homogeneous Linear Equations problem (BHLE) which is introduced in Sect. 4. The formalization of this proof is one of the major achievements in this paper. It posed a significant challenge since it often relied on human intuition and had to be restructured appropriately to allow a formal proof. The main proof steps are explained and difficulties in the formalization effort are described. This proof only works in infinity norm and we explain why. In Sect. 5, the reduction from BHLE to the SVP is given. Again, this proof was quite elaborate to formalize as there were inaccuracies and a lot of intuition was involved. Differences between the formal proof and [7] are explained by examples.

In Sect. 6, we have a quick look at the reduction proof for the CVP in p-norm (for finite $p \geq 1$). In the case of the SVP there only exists a randomized hardness proof in Euclidean norm by Ajtai [1] up to now.

Finally, the time complexity of the reduction functions are considered in Sect. 7. We conclude the paper with a short summary and outlook.

2 Foundations

This section introduces known foundations mainly to fix the terminology and notation: problem reductions, lattices, and the combinatorial problems under consideration (CVP, SVP, Partition and Subset Sum).

2.1 Problem Reductions

Formally, a *decision problem* is given by the set of *YES-instances* P and a set Γ of problem *instances*, where $P \subseteq \Gamma$. We often associate the decision problem with the set of YES-instances, when the instance set Γ is obvious and not explicitly defined. In this paper we will often phrase problems informally (e.g. "decide if p is prime") rather than give them explicitly as sets. For example, the decision problem "decide if a natural number p is prime" will be formalized in the following way: the set of problem instances is $\Gamma = \mathbb{N}$ (in Isabelle these are all elements of type *nat*); and the YES-instances are $P = \{p \in \mathbb{N} \mid p \text{ is prime}\}$ (in Isabelle this is a set of type *nat set*).

Definition 1 (Problem reduction). *Let $A \subseteq \Gamma$ and $B \subseteq \Delta$ be two problems. A function $f : \Gamma \to \Delta$ is a reduction from A to B if it fulfills the following properties:*

- $\forall a \in \Gamma.\ a \in A \Leftrightarrow f(a) \in B$
- *f can be computed in polynomial time*

If A is NP-hard, a reduction to B proves NP-hardness of B.

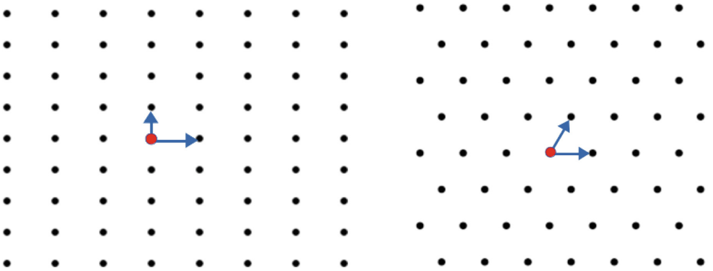
In this paper we present reduction functions informally (e.g. "an a is reduced to a b that is constructed like this") and often with copious amounts of "..." to construct vectors etc. Of course in the formalization these reduction functions are spelled out in complete detail. Since all operations used in the reduction functions in this paper are elementary, the polynomial time property has not been formalized but is briefly discussed in Sect. 7. The focus of our paper are the proofs $a \in A \Leftrightarrow f(a) \in B$.

2.2 Lattice-Based Computational Problems

To have a better understanding, we will first introduce lattices as such. Lattices are a structured set of points. They form an additive, discrete subgroup of $\mathbb{R}^n$. Formally, we define the following.

Definition 2 (Lattice). *Let $A = \{a_1, \ldots, a_n\} \subset \mathbb{R}^n$ be a set of linearly independent vectors. Then the integer span of A forms a lattice $\mathcal{L}$, that is:*

$$\mathcal{L} = \left\{ \sum_{i=1}^{n} c_i a_i \mid c_i \in \mathbb{Z} \right\}$$



(a) Lattice with rectangular basis vectors

(b) Lattice with triangular basis vectors

Fig. 1. Two exemplary lattices in $\mathbb{R}^2$

Example 1. In Fig. 1 two examples of lattices in $\mathbb{R}^2$ are depicted. The red point is the origin. The two blue arrows show the basis vectors a_1 and a_2 that are linearly independent and span the lattice. Every integer combination of the two blue arrows is a black point, an element of the lattice.

We can see that the grid spanned by the basis vectors is discrete and has some recurring structures. These structures are determined by the basis vectors: the angle between them and their length. In Fig. 1a, the angle between the two basis vectors is 90° yielding a rectangular fundamental domain. Whereas in Fig. 1b, we have an angle of 60° between the basis vectors and equal length. This produces a fundamental domain of an equilateral triangle.

Indeed, the automorphism group of a lattice is a symmetry group, see Conway [5, Chapter 3.4]. For example, in Fig. 1a the symmetry group is **pmm** and in Fig. 1b is it **p3m1** [13].

In the rest of the text and in the formalization we restrict to finite bases over $\mathbb{Z}$ (instead of $\mathbb{R}$), simply for computability reasons. Of course bases over $\mathbb{Q}$ can be transformed into bases over $\mathbb{Z}$ by scaling all basis vectors.

The starting point of most known hard problems on lattices are the shortest vector problem and the closest vector problem. They are defined below (as usual in decision and not in optimization form). The lattice $\mathcal{L} \subseteq \mathbb{Z}^n$ is assumed to be generated by a finite basis in $\mathbb{Z}^n$.

Definition 3 (Closest Vector Problem (CVP)). *Given a lattice $\mathcal{L}$, a vector $b \in \mathbb{Z}^n$ and an estimate k, decide whether there exists a vector $v \in \mathcal{L}$ such that*

$$\|v - b\| \leq k$$

Definition 4 (Shortest Vector Problem (SVP)). *Given a lattice $\mathcal{L}$ and an estimate k, determine whether there exists a vector $v \in \mathcal{L}$ such that*

$$\|v\| \leq k \textit{ and } v \neq 0$$

2.3 Partition and Subset Sum Problems

Recall that we plan to prove NP-hardness of the CVP and SVP in the case of the infinity norm by reducing the well-studied NP-complete Subset Sum and Partition problems to the CVP and SVP. We state the definitions.

Definition 5 (Partition problem). *Given a finite list of integers $a_1, \ldots, a_n$, does there exist a partition of $\{1 \ldots n\}$ into subsets I and $\{1 \ldots n\} \setminus I$ such that*

$$\sum_{i \in I} a_i = \sum_{i \in \{1 \ldots n\} \setminus I} a_i$$

The Partition problem can be seen as a special case of the Subset Sum problem.

Definition 6 (Subset Sum problem). *Given a finite list of integers $a_1, \ldots, a_n$ and an integer s, decide whether there exists a subset S of $\{1 \ldots n\}$ such that*

$$\sum_{i \in S} a_i = s$$

2.4 Notation

Throughout the paper we use traditional mathematical notation, in particular the graphical "...". The formal Isabelle notation is by necessity more verbose (and precise). Our formalization employs both lists and vectors as a type for finite sequences and converts between them where necessary. For reasons of presentation we blur this distinction in the paper.

3 CVP

In this section, we formalize the proof of the NP-hardness of the CVP in the infinity norm along the lines of [15, p 48., Chap. 3.2, Thm 3.1] by reducing Subset Sum to the CVP.

An instance $a_1, \dots, a_n, s$ of Subset Sum is mapped to the following instance of the CVP:

$$\mathcal{L} = \begin{pmatrix} a_1 & \cdots & a_n \\ a_1 & \cdots & a_n \\ 2 & & 0 \\ & \ddots & \\ 0 & & 2 \end{pmatrix} \cdot \mathbb{Z}^n \qquad b = \begin{pmatrix} s-1 \\ s+1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \qquad k = 1 \tag{1}$$

We proved the following theorem:

Theorem 1. *The above mapping is a reduction from the Subset Sum problem to the CVP (in infinity norm).*

This implies that the CVP (in infinity norm) is an NP-hard problem.

The reduction function used by Micciancio and Goldwasser [15] actually looks a bit different. The image of $a_1, \dots, a_n, s$ would be

$$B = \begin{pmatrix} a_1 & \cdots & a_n \\ 2 & & 0 \\ & \ddots & \\ 0 & & 2 \end{pmatrix} \qquad \mathcal{L} = B \cdot \mathbb{Z}^n \qquad b = \begin{pmatrix} s \\ 1 \\ \vdots \\ 1 \end{pmatrix} \qquad k = 1 \tag{2}$$

However, the proof in [15, p. 49] with this reduction function works only for $p < \infty$. It goes along the lines of the following idea: Take $k = \sqrt[p]{n}$. In the case of $p = \infty$, we get $k = \lim_{p \to \infty} \sqrt[p]{n} = 1$. Then we can formulate the following equality (equation (3.5) in [15, p. 49]):

$$\|Bx - b\|_p^p = \left| \sum_{i=1}^{n} a_i x_i - s \right|^p + \sum_{i=1}^{n} |2x_i - 1|^p \tag{3}$$

Given a YES-instance $a_1, \dots, a_n, s$ of Subset Sum, there exists a vector $x = (x_1, \dots, x_n) \in \{0,1\}^n$, such that $\sum_{i=1}^{n} a_i x_i - s = 0$ and $|2x_i - 1| = 1$. Then $\|Bx - b\|_p^p = n$ which proves this case.

Given a YES-instance of the CVP defined by $\mathcal{L}$, t and k that are the image of $a_1, \ldots, a_n, s$ under the reduction function as in (2), we get $\|Bx - b\|_p^p \leq n$. Since all values are integers, we have $|2x_i - 1| \geq 1$. It follows that $\sum_{i=1}^n a_i x_i - s = 0$ and $|2x_i - 1| = 1$. Thus, we can deduce that $a_1, \ldots, a_n, s$ was indeed a YES-instance of Subset Sum.

The major problem we encountered was that this proof works fine for $p < \infty$ but for $p = \infty$, the sum in (3) becomes a maximum instead. The equation then reads

$$\|Bx - b\|_\infty = \max \left(\left| \sum_{i=1}^n a_i x_i - s \right|, |2x_i - 1| \text{ for } 1 \leq i \leq n \right)$$

This invalidates the arguments in the proof since $|\sum_{i=1}^n a_i x_i - s|$ can now be in the range $\{-1, 0, 1\}$. The constraints are too lax to ensure the equality to zero.

A solution was to alter the matrix and target vector and add another entry. The matrix and target vector we used are given in Eq. (1). The alternation to $s - 1$ and $s + 1$ forces a linear combination of the a_i to be exactly s in the hardness proof, since $|\sum_i c_i a_i - (s \pm 1)| \leq 1$.

After communicating with Daniele Micciancio, one of the authors of [15], he suggested using a constant $c > 1$ and the generating instance

$$\mathcal{L} = \begin{pmatrix} c \cdot a_1 & \cdots & c \cdot a_n \\ 2 & & 0 \\ & \ddots & \\ 0 & & 2 \end{pmatrix} \cdot \mathbb{Z}^n \qquad b = \begin{pmatrix} c \cdot s \\ 1 \\ \vdots \\ 1 \end{pmatrix} \qquad k = 1$$

This solves the problem as well and can be implemented using e.g. $c = 2$. This technique is described later in the book [15, pp. 49–51] when trying to explain the NP-hardness proof for the SVP in the infinity norm.

3.1 Towards the SVP

The authors of [15] argue that the reduction argument of the SVP can be deduced generating an instance of the SVP using the Subset Sum instance $a_1, \ldots, a_n, s$ in the following way. For $c > 1$, e.g. $c = 2$, take

$$B = \begin{pmatrix} c \cdot a_1 & \cdots & c \cdot a_n & c \cdot s \\ 2 & & 0 & 1 \\ & \ddots & & 1 \\ 0 & & 2 & 1 \end{pmatrix} \qquad \mathcal{L} = B \cdot \mathbb{Z}^{n+1} \qquad k = 1$$

The authors claim that every shortest vector in the image of the reduction function has -1 as last coefficient. For example, let a YES-instance of the SVP be defined by the generating matrix B of the lattice and let $x = (x_1, \ldots, x_n, -1)^T$

be the coefficients such that Bx is a shortest vector. Then we know that

$$\|Bx\|_\infty = \left\| \begin{pmatrix} c \cdot (x_1 a_1 + \cdots + x_n a_n - s) \\ 2x_1 - 1 \\ \vdots \\ 2x_n - 1 \end{pmatrix} \right\|_\infty \leq 1$$

Since $c > 1$, it follows, that $x_1 a_1 + \cdots + x_n a_n - s = 0$, which yields a solution for the given Subset Sum instance $a_1, \ldots, a_n, s$.

However, this reduction does not always work as the following example shows:

Example 2. Given the Subset Sum instance $(a_1, a_2, a_3, s) = (1, 1, 1, 1)$. This is a YES-instance, since a solution is given by $x_1 = 1$, $x_2 = 0$ and $x_3 = 0$. The basis matrix of the corresponding SVP would be (with $c > 1$)

$$B = \begin{pmatrix} c & c & c & c \\ 2 & 0 & 0 & 1 \\ 0 & 2 & 0 & 1 \\ 0 & 0 & 2 & 1 \end{pmatrix}$$

Take for example the vector $v = B \cdot (-1, -1, -1, 3)^T = (0, 1, 1, 1)^T$. It has infinity norm 1 and is thus a shortest vector in the lattice generated by B. However, this vector has the last coefficient 3 and not -1, even though it clearly is a shortest vector of the lattice given by B. The corresponding scaled "solution" for Subset Sum would be $(1/3, 1/3, 1/3, -1)$ but since only integer values are allowed in the solution space, this is not a solution in our sense.

We consider another example. Let the Subset Sum instance be $a'_1 = 3, s' = 1$. We can easily see that this is not a YES-instance, i.e. there exists no solution. Still, the corresponding SVP instance given via the reduction function is generated by the matrix

$$B' = \begin{pmatrix} c \cdot 3 & c \cdot 1 \\ 2 & 1 \end{pmatrix}$$

In this case the coefficients $(-1, 3)^T$ yield a shortest vector in the lattice spanned by B', since

$$\left\| B' \begin{pmatrix} -1 \\ 3 \end{pmatrix} \right\|_\infty = \left\| \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\|_\infty \leq 1$$

Thus, B' defines a YES-instance of the SVP, but the original Subset Sum instance is not a YES-instance.

In [15], it is stated for the infinity norm that any shortest vector yields a solution for the Subset Sum Problem, which is not the case in these examples: we cannot ensure that a shortest vector always has -1 as a last coordinate.

Although the proof in [15] does not work out as expected, there is still the reduction proof by van Emde-Boas [7] which reduces a problem called the Bounded Homogeneous Linear Equation problem to the SVP in infinity norm. This will be discussed in the next two sections.

4 Bounded Homogeneous Linear Equations

A technical report by Peter van Emde-Boas [7] gives another reduction proof for the NP-hardness of the SVP in infinity norm. The author first reduces the Partition Problem to a problem called Bounded Homogeneous Linear Equation (BHLE) which is then reduced to the SVP.

Definition 7 (Bounded Homogeneous Linear Equations problem). *Given a finite vector of integers $b \in \mathbb{Z}^n$ and a positive integer k, decide whether there exists an $x \in \mathbb{Z}^n \setminus \{0\}$ with $\|x\|_\infty \leq k$ such that*

$$\langle b, x \rangle = 0$$

We have verified a reduction from Partition to BHLE, and thus BHLE is NP-hard.

Theorem 2. *There is a reduction from Partition to BHLE in infinity norm.*

The proof is carefully engineered and rather intricate. Differences to the original proof and problems encountered during the formalization are:

- Our formal proof has a different structure than the proof in the technical report [7]. Indeed, the technical report first proves the reduction of a weaker form of Partition to BHLE and then argues that "omitting" an element yields the desired result as it adds stricter constraints. In the formalization we skip this intermediate step and directly prove the existence of an appropriate reduction function.
- Steps that seem trivial in the technical report often require a long formal proof. What can be reasoned by intuition in a pen-and-paper proof has to be elaborated in the formal proof. Intuition is also sometimes used for hand-waving over small gaps or imprecisions.
- Indexing vectors and lists has been a problem in the formalization. In pen-and-paper proofs, one can argue easily about "omitting" an element of a list even though this is imprecise and often misuses the notation. In the formalization one cannot simply skip an index. All indexing functions in the formalization have to be total. "Omitting" an element can only be solved by re-indexing and re-structuring the lists in the proof.
- Numbers are interpreted in different number systems during the proof. In contrast to the original proof, the formalization has to explicitly state the digits for a change of basis and show equivalence. This leads to verbose and elaborate proofs. To make proofs easier, we use the concrete basis $d = 5$ instead of an unspecified basis $d > 4$ as in [7]. Furthermore, the number M must use the absolute values of the a_i (omission in the definition of M in [7]). The formal definition is stated below.
- The proof involved many arguments about manipulations of huge sums. Working with huge sums entails very large proof states where the existing proof automation mostly failed on. These proof states require detailed (but still readable) proofs and occasional manual instantiation of theorems. Another possible solution to get smaller proof states is to introduce local abbreviations for subterms.

Let us have a look at the proof and its difficulties in the formalization in more detail. We start from a Partition instance $a = a_1, \dots, a_n$. Note that we ignore the trivial case $n = 0$ in this presentation (but deal with it in the formal proofs)—this means $n - 1 \geq 0$. We reduce a to a BHLE instance b as follows:

- Define

$$M = 2 \cdot (\sum_{i=1}^{n} |a_i|) + 1 \tag{4}$$

- For $1 \leq i < n$ generate a 5-tuple

$$\begin{aligned} b_{i,1} &= a_i + M \cdot (5^{4i-4} + 5^{4i-3} + 5^{4i-1}) \\ b_{i,2} &= M \cdot (5^{4i-3} + 5^{4i}) \\ b_{i,3} &= M \cdot (5^{4i-4} + 5^{4i-2}) \\ b_{i,4} &= a_i + M \cdot (5^{4i-2} + 5^{4i-1} + 5^{4i}) \\ b_{i,5} &= M \cdot (5^{4i-1}) \\ b_i &= b_{i,1}, b_{i,2}, b_{i,4}, b_{i,5}, b_{i,3} \end{aligned} \tag{5}$$

  Note that $b_{i,3}$ has moved to the last position in b_i.
- For $i = n$ generate only a 4-tuple:

$$\begin{aligned} b_{n,1} &= a_n + M \cdot (5^{4n-4} + 5^{4n-3} + 5^{4n-1}) \\ b_{n,2} &= M \cdot (5^{4n-3} + 1) \\ b_{n,4} &= a_n + M \cdot (5^{4n-2} + 5^{4n-1} + 1) \\ b_{n,5} &= M \cdot (5^{4n-1}) \\ b_n &= b_{n,1}, b_{n,2}, b_{n,4}, b_{n,5} \end{aligned} \tag{6}$$

  Note that
  - $b_{n,3}$ is omitted from b_n to restrict the constraints necessary for the proof and
  - that in $b_{n,2}$ and $b_{n,4}$ the last summand changes to a +1 in comparison to the other $b_{i,2}$ and $b_{i,4}$.

In summary, the entry $b_{i,3}$ is uniformly in the last position in the b_i but omitted from the final b_n.

The Partition instance a of length n is reduced to a vector b of length $5n - 1$:

$$b = (b_1, \dots, b_{n-1}, b_n) \tag{7}$$

The NP-hardness proof now follows in three steps:

1. We need to show an auxiliary lemma.
2. We show that a YES-instance of Partition is reduced to a YES-instance of BHLE.
3. We show that the pre-image of a YES-instance of BHLE is indeed a YES-instance in Partition.

4.1 Auxiliary Lemma

As a first step, the proof needs a short auxiliary lemma from number theory.

Lemma 1. *Let $x, y, c \in \mathbb{Z}^n$ and M be an integer. Assume that $M > \sum_{i=1}^{n} |x_i|$ and that $|c_i| \leq 1$ for all $1 \leq i \leq n$. Furthermore, let the following equation hold:*

$$\sum_{i=1}^{n} c_i \cdot (x_i + M \cdot y_i) = 0 \tag{8}$$

Then we have

$$\langle c, x \rangle = 0 \quad \text{and} \quad \langle c, y \rangle = 0$$

In this lemma, we can reinterpret $x_i + M \cdot y_i$ from (8) as a number in basis M with lowest digit x_i. Even with a coefficient c_i, the lowest digit in basis M has to be zero, as well as the rest. By splitting off the lowest digits consecutively, we can show, that indeed all digits in basis M have to equal zero.

4.2 $a \in$ Partition $\Longrightarrow$ $b \in$ BHLE

This direction is quite easy. Let $a_1, \ldots, a_n$ be a YES-instance of partition with partitioning set I. We will show that the following vector x is a solution to the corresponding BHLE:

$$x = (x_1, \ldots, x_{n-1}, x_n)$$

$$x_i = \begin{cases} 1, -1, 0, -1, 0 & i \in I \wedge n-1 \in I \\ 0, 0, -1, 1, 1 & i \in I \wedge n-1 \notin I \\ 0, 0, -1, 1, 1 & i \notin I \wedge n-1 \in I \\ 1, -1, 0, -1, 0 & i \notin I \wedge n-1 \notin I \end{cases} \qquad 1 \leq i < n$$

$$x_n = 1, -1, 0, -1$$

We have to show that $\langle b, x \rangle = 0$. This is proven by plugging in the definitions and rearranging terms in the sum of the scalar product such that they cancel out. As a last step in the proof, we need to show that $\|x\|_\infty \leq 1$. For the infinity norm this is quite easy. However, it would not be true for other norms. For $p \geq 1$ and $p < \infty$ we have for $n \geq 1$:

$$\|x\|_p = \sqrt[p]{3n} > 1$$

Thus, the chosen constraints x only work in infinity norm.

4.3 $a \in$ Partition $\Longleftarrow$ $b \in$ BHLE

This direction is harder. Let b be a YES-instance of BHLE. That is, there exists a nonzero x such that $\langle b, x \rangle = 0$ and $\|x\|_\infty \leq 1$. We have to show that there is a partition I on $a_1, \ldots, a_n$ with $\sum_{i \in I} a_i = \sum_{i \in \{1 \ldots n\} \setminus I} a_i$.

The proof idea works as follows. First, we apply the auxiliary lemma and get a constraint on the a_i on the one hand, and a condition on the x_i with coefficients that are powers of 5 on the other hand. Using this condition on the x_i, we generate equational constraints on the entries of x by looking at the digits in basis 5. We argue that a number equals zero if and only if all its digits are zero.

The generated equations lead to a good characterisation of x, namely the weight $w = x_{5(n-1)+1}$. From the assumption that $\|x\|_\infty \leq 1$, we deduce $|w| \leq 1$. Again, this step can only be reasoned in the infinity norm. For other p-norms, this argumentation breaks as we need the property $|w| \leq 1$ to complete the proof. Using the value of w, we can constuct a partitioning set I with the required property from the equation on the a_i.

5 SVP

Knowing that the BHLE is indeed an NP-hard problem, we reduce it to the SVP. Then we can conclude that the SVP in infinity norm is NP-hard.

Theorem 3. *There is a reduction from BHLE to the SVP in infinity norm.*

Again some difficulties were met when formalizing the proof for the above theorem. First of all, note that the terminology in [7] and nowadays is a bit different. In [7], the shortest vector problem only denotes the shortest vector problem in the Euclidean norm. What we call the shortest vector problem in the infinity norm is named closest vector problem in [7]. To make terminology even more confusing, our understanding of the closest vector problem is called the nearest vector problem in [7]. To make the notation clear, we provide a table for reference in Fig. 2.

technical report [7]	our notation
closest vector problem	SVP in infinity norm
shortest vector problem	SVP in Euclidean norm
nearest vector problem	CVP

Fig. 2. Notation

A more mathematical problem encountered was that the reduction itself used in [7] was not entirely correct. In the reduction two factors $k' = k+1$ and k'' were introduced. These factors should have certain properties to allow the arguments of the reduction proof to go through. However, this is only true when tweaking these factors a bit to make the whole proof watertight. We will now have a closer look.

Given the BHLE instance $b = (b_1, \dots, b_n)$ and k, create the following SVP instance:

$$\mathcal{L} = \begin{pmatrix} 1 & & 0 & 0 \\ & \ddots & & \vdots \\ 0 & & 1 & 0 \\ \multicolumn{3}{l}{-\ (k+1) \cdot b} & -\ k'' \end{pmatrix} \cdot \mathbb{Z}^n \qquad k = k$$

where k'' is the factor in question. In the technical report, we have

$$k'' = 2 \cdot (k+1) \cdot (\sum_i b_i) + 1$$

The following example however shows that this factor is not enough.

Example 3. Consider the BHLE instance given by $b = (1, -1)$ and $k = 1$. This is a YES-instance, since the vector $(1, 1)$ yields the expected properties.

Define the following matrices.

$$B_0 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & -2 & 1 \end{pmatrix} \qquad B_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & -2 & 9 \end{pmatrix} \qquad B_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 6 & -6 & 25 \end{pmatrix}$$

The associated SVP instance is the lattice generated by B_0. Then the vector $(0, 0, 1)^T$ with infinity norm 1 is a solution to the SVP instance generated by the basis matrix B_0. However, since the last entry is nonzero, this does not provide a solution for BHLE. Contrary to this example, the proof in the technical report shows that for all SVP solutions the last entry must be zero.

The reason, why the argument in the technical report breaks at this point is because $b_1 + b_2 = 0$, thus making $k'' = 1$ very small. One step to prevent this is to use the absolute values of the b_i in k'' instead. The new k''_1 we consider is

$$k''_1 = 2 \cdot (k+1) \cdot (\sum_i |b_i|) + 1$$

With this new factor k''_1 we get the generating matrix B_1 and the vector $(0, 0, 1)$ is no longer a shortest vector.

Still, this is not enough. Consider the same $b = (1, -1)$ as above, but let $k = 5$. Then we get B_2 as the generating matrix of the SVP lattice. The vector $x = (0, 5, 1)^T$ is a shortest vector whose last entry is nonzero. Again it contradicts the proof in the technical report. The reason this time is the following: the argument that $(k+1)\left(\sum_{i=1}^n x_i b_i\right)$ and k''_1 have different relative sizes fails. Indeed, we have

$$\left\| \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 6 & -6 & 25 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 5 \\ 1 \end{pmatrix} \right\|_\infty = \left\| \begin{pmatrix} 0 \\ 5 \\ -5 \end{pmatrix} \right\|_\infty = 5 \le k$$

We can obtain different relative sizes of $(k+1)\left(\sum_{i=1}^n x_i b_i\right)$ and k''_1 by defining

$$k''_2 = 2 \cdot k \cdot (k+1) \cdot (\sum_i |b_i|) + 1 \tag{9}$$

Now we can make sure that the last entry of a solution to the SVP problem is indeed zero. For the proof of Theorem 3 we consider the reduction given by

$$\mathcal{L} = \underbrace{\begin{pmatrix} 1 & & 0 & 0 \\ & \ddots & & \vdots \\ 0 & & 1 & 0 \\ - & (k+1) \cdot b & - & k_2'' \end{pmatrix}}_{B} \cdot \mathbb{Z}^n \qquad k = k$$

where B denotes the basis matrix generating the lattice $\mathcal{L}$ as given above.

Consider a solution $x = (x_1, \ldots, x_{n+1})$ of the SVP with $\|Bx\|_\infty \leq k$. Then we have

$$Bx = \begin{pmatrix} 1 & & 0 & 0 \\ & \ddots & & \vdots \\ 0 & & 1 & 0 \\ - & (k+1) \cdot b & - & k_2'' \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ \vdots \\ x_n \\ x_{n+1} \end{pmatrix} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \\ (k+1)(\sum_{i=1}^n x_i b_i) + x_{n+1} \cdot k_2'' \end{pmatrix}$$

As this yields a solution to the SVP, we get:

$$|(k+1)(\sum_{i=1}^n x_i b_i) + x_{n+1} \cdot k_2''| \leq k \tag{10}$$

Then we calculate:

$$\begin{aligned} (k+1)(\sum_{i=1}^n x_i b_i) + x_{n+1} \cdot k_2'' \leq (k+1)(\sum_{i=1}^n |x_i||b_i|) + x_{n+1} \cdot k_2'' \leq \\ \leq (k+1)k(\sum_{i=1}^n |b_i|) + x_{n+1} \cdot k_2'' \end{aligned}$$

Assuming that $x_{n+1} \neq 0$, we have

$$|(k+1)k(\sum_{i=1}^n |b_i|)| < |2 \cdot k \cdot (k+1) \cdot (\sum_i |b_i|) + 1| = |k_2''| \leq |x_{n+1} \cdot k_2''|$$

Thus the two summands indeed have different relative sizes and can never cancel out the other summand. This leads to a contradiction to (10). Therefore, $x_{n+1} = 0$ must be true and $(x_1, \ldots, x_n)$ constitutes a solution to the BHLE when using k_2'' as in (9).

6 Other *p*-Norms

Up to now, we have investigated lattice problems under the infinity norm. Even though this yields nice hardness results, in practice the Euclidean norm is used more often. Unfortunately, when considering p-norms things do not play out as nicely. In this section, we assume $1 \leq p < \infty$ whenever we talk about a specific p.

For the CVP, there is a generalisation of the proof for every p-norm in [15, p. 48, Chap. 3.2, Thm 3.1] which we also formalized. Let $a_1, \ldots, a_n, s$ be an instance of Subset Sum. The reduction function maps this instance to:

$$\mathcal{L} = \begin{pmatrix} a_1 & \cdots & a_n \\ 2 & & 0 \\ & \ddots & \\ 0 & & 2 \end{pmatrix} \cdot \mathbb{Z}^n \qquad b = \begin{pmatrix} s \\ 1 \\ \vdots \\ 1 \end{pmatrix} \qquad k = \sqrt[p]{n}$$

Then the following theorem holds:

Theorem 4. *The above mapping is a reduction from the Subset Sum problem to the CVP in p-norm.*

This implies that the CVP in p-norm is an NP-hard problem. The outline to the proof is given in Sect. 3 after Theorem 1. The important difference to the infinity norm is that the bound k scales with the dimension n of the lattice.

For the SVP, there is no known deterministic NP-hardness result in the Euclidean norm, or even any p-norm. However, Ajtai [1,2] found an interesting alternative which is quite useful for the application in cryptography, namely randomized reductions using polynomial-time probabilistic reduction functions. In cryptography, these results guarantee the hardness of "average" cases. That is, given an average instance according to a probability distribution, it will most likely be intractable.

7 Time Complexity

As stated in Sect. 2, time complexity of the above reduction functions has not been formalized. However, we give a short explanation why all reduction functions are indeed in polynomial time.

Subset Sum to CVP: The reduction function as given in Eq. (1) creates $(n+2)(n+1)+1$ values using only memory access or one addition. Therefore, the time complexity in this case is $\mathcal{O}(n^2)$.

Partition to BHLE: In this case, the reduction function maps the input a of length n to b as defined in Eq. (7). The value $k = 1$ is fixed. Then a is mapped to a vector of length $5n-1$. When calculating the b_i, we need to calculate the value of M as in (4). As we sum over all input values, this lies in $\mathcal{O}(n)$. Each b_i can then be calculated in $\mathcal{O}(n)$ since it only contains a constant number of additions of the input with fixed cofactors (see (5)–(6)). Putting the construction of the list and the calculation of the b_i together, we find that the whole reduction function is in $\mathcal{O}(n^2)$.

BHLE to the SVP: Consider the reduction function as given in Eq. (5) using the value k_2'' as in (9). Calculating k_2'' requires $n+2$ memory accesses which are processed in $n+4$ arithmetic operations, thus having a time complexity of $\mathcal{O}(n)$. Every other entry in the matrix is calculated on $\mathcal{O}(1)$, since they contain at most two memory accesses and at most two arithmetic operations. The input generates $(n+1)^2+1$ values, of which $(n+1)(n+1)$ are in $\mathcal{O}(1)$ (namely all the zeros and ones, the vector $(k+1)\cdot a$ and the constraint k) and one is calculated in $\mathcal{O}(n)$ (namely k_2''). Thus, the whole reduction function lies in $\mathcal{O}(n^2)$.

8 Outlook

With this paper, we now have a formal proof for NP-hardness of the CVP and SVP in the infinity norm, as well as a formal proof of the CVP in p-norm (for $1 \leq p < \infty$). In the formalization process, many gaps and imprecisions in the pen-and-paper proofs were fixed. The changes to the original proofs have been elaborated with explanations and examples. Unfortunately, giving a deterministic reduction proof of the SVP in p norm for $p < \infty$ is still an open problem. Under probabilistic assumptions, Ajtai showed NP-hardness of the SVP in Euclidean norm in [2].

An interesting topic for future work is to develop a framework for probabilistic reductions such as in [2]. This will give the foundation to extend formalization of hardness proofs to other problems in lattice theory, especially those used in lattice-based cryptography, such as the Learning with Errors (LWE) Problem, Ring-LWE and Module-LWE. This will underline the security of many lattice-based crypto systems. Another topic for future work is to formalize the hardness proofs for approximate versions of the CVP and SVP.

Acknowledgements. We thank Manuel Eberl for continuous support and fruitful discussions. The first author gratefully acknowledges the financial support of this work by the research training group ConVeY funded by the German Research Foundation under grant GRK 2428.

References

1. Ajtai, M.: Generating hard instances of lattice problems. Electron. Colloquium Comput. Complex. **3** (1996)
2. Ajtai, M.: The shortest vector problem in L2 is NP-hard for randomized reductions (extended abstract). In: Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing, STOC 1998, Dallas, Texas, USA, pp. 10–19. ACM Press (1998)
3. Babai, L.: On Lovász' lattice reduction and the nearest lattice point problem. Combinatorica **6**, 1–13 (1986)
4. Balbach, F.J.: The Cook-Levin theorem. Archive of Formal Proofs (2023). https://isa-afp.org/entries/Cook_Levin.html. Formal proof development
5. Conway, J.H., Sloane, N.J.A.: Sphere Packings, Lattices and Groups. Springer, New York (1999). https://doi.org/10.1007/978-1-4757-6568-7
6. Dinur, I., Kindler, G., Raz, R., Safra, S.: Approximating CVP to within almost-polynomial factors is NP-hard. Combinatorica **23**, 205–243 (2003). https://doi.org/10.1007/s00493-003-0019-y
7. van Emde Boas, P.: Another NP-complete partition problem and the complexity of computing short vectors in a lattice. Technical report 81-04. Technical report, Mathematisch Instituut, Roetersstraat 15, 1018 WB Amsterdam, The Netherlands (1981)

8. Gäher, L., Kunze, F.: Mechanising complexity theory: the Cook-Levin theorem in coq. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2021). https://doi.org/10.4230/LIPICS.ITP.2021.20. https://drops.dagstuhl.de/opus/volltexte/2021/13915/
9. Haviv, I., Regev, O.: Tensor-based hardness of the shortest vector problem to within almost polynomial factors. In: Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing, STOC 2007, pp. 469–477. Association for Computing Machinery, New York (2007)
10. Khot, S.: Hardness of approximating the shortest vector problem in lattices. J. ACM **52**(5), 789–808 (2005)
11. Kreuzer, K.: Hardness of lattice problems. Archive of Formal Proofs (2023). https://isa-afp.org/entries/CVP_Hardness.html. Formal proof development
12. Lenstra, A.K., Lenstra, H., Lovasz, L.: Factoring polynomials with rational coefficients. Math. Ann. **261**, 515–534 (1982)
13. Liu, Y., Collins, R.: Frieze and wallpaper symmetry groups classification under affine and perspective distortion. Technical report. CMU-RI-TR-98-37, Carnegie Mellon University, Pittsburgh, PA (1998)
14. Micciancio, D.: The shortest vector in a lattice is hard to approximate to within some constant. In: Proceedings 39th Annual Symposium on Foundations of Computer Science (Cat. No. 98CB36280), pp. 92–98 (1998). https://doi.org/10.1109/SFCS.1998.743432
15. Micciancio, D., Goldwasser, S.: Complexity of Lattice Problems. Springer, Boston (2002)
16. Micciancio, D., Regev, O.: Lattice-based cryptography. In: Bernstein, D.J., Buchmann, J., Dahmen, E. (eds.) Post-Quantum Cryptography, pp. 147–191. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-540-88702-7_5
17. Nipkow, T., Klein, G.: Concrete Semantics with Isabelle/HOL. Springer, Cham (2014). http://concrete-semantics.org
18. Nipkow, T., Wenzel, M., Paulson, L.C. (eds.): Isabelle/HOL—A Proof Assistant for Higher-Order Logic. LNCS, vol. 2283. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45949-9
19. Rothvoss, T., Venzin, M.: Approximate CVP in time $2^{0.802n}$ – now in any norm! arXiv:2110.02387 [cs] (2021)