格子簡約アルゴリズムを用いたナップサック暗号への攻撃についての考察
Lattice Reduction Attack on the Knapsack Type Cryptosystem

大庭 翔介 † ツォイクマン トーマス †
Shosuke Oba Thomas Zeugmann

Abstract

The *subset-sum problem* is, for given positive integers $a_0, \ldots, a_{n-1}$, and M, to find a subset of the a_i that sums up to M. There are known *low-density attacks* enabling us to solve almost all subset-sum problems of *density* $d < 0.9408$ in polynomial time. In this paper, we review these methods and perform computer experiments to compare algorithms to solve subset-sum problems using the L^3-algorithm.

1. Introduction

Knapsack cryptosystems are cryptosystems whose security is based on the difficulty of the *subset-sum problem*. Merkle–Hellman knapsack cryptosystem is one of the first practical public key cryptosystems. It is not used anymore, because it has been broken.

Lagarias and Odlyzko [2] proposed an attack for the Merkle–Hellman knapsack cryptosystem in 1985. They showed that almost all subset-sum problems of *density* $d < 0.645$ are efficiently solvable. Later, Coster et al. [1] improved this method.

They reduced subset-sum problems to the problem to find a shortest vector in a lattice (SVP). Here, lattices are sets of all integral combinations of linearly independent vectors. SVP is NP-hard, but by using the *L^3-algorithm*, proposed by Lenstra et al. [3], SVP can be approximated in polynomial time.

In [2], the authors confirmed that their method behaves well using the L^3-algorithm. However, in their experiments the dimension was at most 50 because the performance of computers at that time was not as high as current. In this paper, we compare these algorithm with L^3-algorithm in higher dimensions.

In Section 2, we recall some basic knowledge of subset-sum problems and lattices. In Section 3, we recall previous results by Lagarias and Odlyzko[2] and by Coster et al. [1]. In Section 4, a computer experiment is done to compare these algorithms, where we use the L^3-algorithm to compute a reduced basis.

2. Preliminaries

We recall some basic knowledge about the subset-sum problem and lattices.

† 北海道大学 大学院 情報科学研究科
Graduate School of Information Science and Technology, Hokkaido University

2.1. Subset-Sum Problems

Definition 1. *The subset-sum problem is the problem, given $a_0, \ldots, a_{n-1} \in \mathbb{N}$ and $M \in \mathbb{N}$ as inputs, to find $x_0, \ldots, x_{n-1} \in \{0, 1\}$ such that $\sum_{i=0}^{n-1} x_i a_i = M$.*

We call $a_0, \ldots, a_{n-1}$ in Definition 1 *weights*. The subset-sum problem is known to be NP-complete.

Definition 2. *For a subset-sum problem with weights $a_0, \ldots, a_{n-1}$, its density d is defined by*

$$d := n / \log_2(\max_i a_i).$$

Regarding the density, it is known that if a_i are chosen at random, when density is significantly larger than 1 then the subset-sum problem has many solutions in general, and when the density is approximately 1 then the most difficult problem arise.

2.2. Lattices

Definition 3. *Let $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in \mathbb{R}^m$ be n linearly independent vectors. Then, a lattice generated from these vectors is defined as*

$$L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) := \left\{ \sum_{i=1}^{n} x_i \boldsymbol{b}_i \mid x_i \in \mathbb{Z}^m \right\}.$$

In Definition 3, the vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ are called a *basis* of the lattice.

The L^3-algorithm is a polynomial time algorithm to transform the input basis to a basis called *LLL reduced basis*. These input and output bases generate the same lattice. Regarding LLL reduced bases, the following theorem is important:

Theorem 1. *If $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ is a LLL reduced basis of a lattice L with parameter $1/4 < \delta < 1$, then*

$$\|\boldsymbol{b}_1\| \leq (2/\sqrt{4\delta - 1})^{n-1} \min_{\boldsymbol{x} \in L \setminus \{0\}} \|\boldsymbol{x}\|.$$

This theorem states that the first basis vector in an LLL reduced basis is an approximation the shortest vector in the lattice up to an exponential factor.

3. Previous Results

3.1. Lagarias and Odlyzko's Method

Lagarias and Odlyzko [2] proposed an algorithm to solve low-density subset-sum problems. Their approach was as follows: For given $a_0, \ldots, a_{n-1}$, and M,

we form the lattice with the basis:

$$\begin{pmatrix} \boldsymbol{b}_0 \\ \boldsymbol{b}_1 \\ \vdots \\ \boldsymbol{b}_{n-1} \\ \boldsymbol{b}_n \end{pmatrix} := \begin{pmatrix} 1 & 0 & \cdots & 0 & Na_0 \\ 0 & 1 & \cdots & 0 & Na_1 \\ & & \ddots & & \vdots \\ 0 & 0 & \cdots & 1 & Na_{n-1} \\ 0 & 0 & \cdots & 0 & NM \end{pmatrix}, \quad (1)$$

where N is an integer such that $N > \sqrt{n/2}$. If the subset-sum problem has a solution $x_0, \ldots, x_{n-1}$, the lattice has a vector

$$\boldsymbol{x} := \sum_{i=0}^{n-1} x_i \boldsymbol{b}_i + \boldsymbol{b}_n = (x_0, \ldots, x_{n-1}, 0).$$

Furthermore, it is considerably short. Lagarias and Odlyzko [2] proved the following theorem:

Theorem 2. *Given subset-sum problem instance, if $a_0, \ldots, a_{n-1}$ are randomly chosen and the density $d < 0.645$, above lattice has a vector $\boldsymbol{x}$ as a shortest nonzero vector with sufficiently high probability.*

We can use L^3-algorithm to find a shortest nonzero vector in a lattice. We can find the solution of subset-sum problems by finding the vector $\boldsymbol{x}$ in the LLL reduced basis.

3.2. CJLOSS Method

Coster et al. [1] improved Lagarias and Odlyzko's [2] method. They made important changes to the lattice basis shown in (1). The vector $\boldsymbol{b}_n$ is replaced by

$$\boldsymbol{b}_n = (1/2, 1/2, \ldots, 1/2, NM)\,,$$

where N is an integer such that $N > \sqrt{n}/2$. If a subset-sum problem has a solution $x_0, \ldots, x_{n-1}$, the lattice has the vector

$$\hat{\boldsymbol{x}} := \sum_{i=0}^{n-1} x_i \boldsymbol{b}_i + \boldsymbol{b}_n = (x_0 - 1/2, \ldots, x_{n-1} - 1/2, 0)\,.$$

The length of the vector is $\sqrt{n}/2$. It is also considerably short. They proved the following theorem:

Theorem 3. *Given subset-sum problem instance, if $a_0, \ldots, a_{n-1}$ are randomly chosen and the density $d < 0.9408$, above lattice has a vector $\hat{\boldsymbol{x}}$ as a shortest nonzero vector with sufficiently high probability.*

4. Computer Experiments

We performed a computational test to compare algorithms in Section 3: Lagarias–Odlyzko algorithm, and the CJLOSS algorithm. We used an Intel(R) Core^{TM} i7-2620M CPU @ 2.70GHz and NTL 10.3.0 [4] for the L^3-algorithm.

In the tests we fixed the dimension n of subset-sum problems and the bit length b of weights. Then we generated $a_0, \ldots, a_{n-1} \in \{1, \ldots, 2^b\}$ uniformly at random, and generated $x_0, \ldots, x_{n-1} \in \{0, 1\}$ at random such that $x_i = 1$ for exactly $n/2$ of the x_i's. For generated subset-sum problems, we constructed lattice bases, and then we permuted the order of vectors in the bases. After that, we ran L^3-algorithm on the bases. By permuting the order of vectors in bases, the L^3-algorithm outputs different reduced bases. If we found a desired form vector in the basis, the run was a success and halted. If we could not find a solution, repeat these steps at most 10 times.

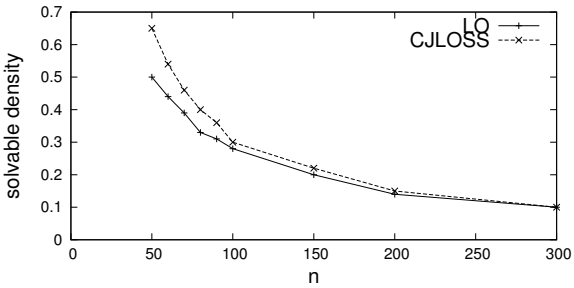

Figure1 Comparing the LO and CJLOSS algorithms

The results are in Fig. 1. As the dimension was higher, the density of solved subset-sum problems decreased in both of algorithms. This is mentioned in [2], and they conjectured the density of solved subset-sum problems goes to 0 as n goes to infinity. This result also suggests it. Success rates of the CJLOSS algorithm are higher than of the Lagarias-Odlyzko algorithm, as well in high dimensions.

In all results, the density of solved subset-sum problems fall much below the theoretical bound in high dimensions. The density of almost all of subset-sum problems are solved seems to be in inverse proportion to dimension. It is considered that the L^3-algorithm can find much short vectors in low dimensions, but its performance degrades as the dimension goes higher. To find a shortest vector, further devising may be needed.

References

[1] M. J. Coster, A. Joux, B. A. LaMacchia, A. M. Odlyzko, C.-P. Schnorr, and J. Stern. Improved low-density subset sum algorithms. *computational complexity*, 2(2):111–128, 1992.

[2] J. C. Lagarias and A. M. Odlyzko. Solving low-density subset sum problems. *Journal of the Association for Computing Machinery*, 32(1):229–246, Jan. 1985.

[3] A. Lenstra, H. Lenstra, and L. László. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261:515–534, 1982.

[4] V. Shoup. A library for doing number theory. version 10.3.0. `http://www.shoup.net/ntl`.