Solving Random Low-Density Subset Sum Using Babai's Algorithm

Huck Bennett*

July 21, 2024

Abstract

Motivated by the goal of breaking knapsack cryptography including the Merkle-Hellman cryptosystem [MH78], Lagarias and Odlyzko [LO85] gave an algorithm for solving random low-density Subset Sum problems using lattice techniques. Frieze [Fri86] later gave a simplified analysis of their main result. The goal of this note is to simplify this analysis further. The main idea behind the additional simplification is to reduce the input random Subset Sum instance to approximate CVP instead of approximate SVP.

The term *knapsack cryptography* is used to describe a class of cryptosystems that use functions of the form $\boldsymbol{x} \mapsto \langle \boldsymbol{x}, \boldsymbol{a} \rangle$ to encrypt messages $\boldsymbol{x} \in \{0,1\}^n$, where $\boldsymbol{a} \in (\mathbb{Z}^+)^n$ is the public key. The task of recovering $\boldsymbol{x}$ from $\boldsymbol{a}$ and $s := \langle \boldsymbol{x}, \boldsymbol{a} \rangle$ corresponds to solving an instance of the *Subset Sum* problem, which we next define formally.

Definition 1. The search version of Subset Sum is the problem defined as follows. Given $\boldsymbol{a} \in (\mathbb{Z}^+)^n$ and $s := \langle \boldsymbol{x}, \boldsymbol{a} \rangle = \sum_{i=1}^n x_i a_i$ for some $\boldsymbol{x} \in \{0,1\}^n$ as input, output $\boldsymbol{x}' \in \{0,1\}^n$ such that $s = \langle \boldsymbol{x}', \boldsymbol{a} \rangle$.

Of course, to be useful for cryptography the encryption function $\boldsymbol{x} \mapsto \langle \boldsymbol{x}, \boldsymbol{a} \rangle$ must be injective, $\boldsymbol{x}$ must be efficiently recoverable from s given a trapdoor, and $\boldsymbol{x}$ must not be efficiently recoverable from s and $\boldsymbol{a}$ without the trapdoor. We refer the reader to [Pei15] for a discussion of these issues and focus here on proving the main result of Lagarias and Odlyzko, which gives an efficient algorithm for solving low-density average-case Subset Sum.[1]

Theorem 2 ([LO85, Fri86])**.** *Let n and N be positive integers with $N \geq 2^{(1/2+\varepsilon)n^2}$ for some constant $\varepsilon > 0$, let $a_1, \ldots, a_n \sim \{1, \ldots, N\}$ be sampled independently and uniformly at random, let $\boldsymbol{a} := (a_1, \ldots, a_n)^T$, let $\boldsymbol{x} \in \{0,1\}^n$, and let $s := \langle \boldsymbol{x}, \boldsymbol{a} \rangle$. There is a deterministic polynomial-time algorithm that, given $\boldsymbol{a}$ and s as input, outputs $\boldsymbol{x}$ with overwhelming probability (over the choice of $\boldsymbol{a}$).*

Proof. The algorithm works as follows. First, it outputs the following instance of approximate CVP:

$$B = \begin{pmatrix} \beta a_1 & \beta a_2 & \cdots & \beta a_n \\ 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \in \mathbb{Z}^{(n+1)\times n} , \qquad \boldsymbol{t} = \begin{pmatrix} \beta s \\ 1/2 \\ 1/2 \\ \vdots \\ 1/2 \end{pmatrix} \in \mathbb{Z}^{n+1} , \tag{1}$$

where $\beta := \lceil (n+1) \cdot 2^{n/2} \rceil$ is a large scaling factor. Second, it runs the LLL-algorithm [LLL82] on B with $\delta = 3/4$, receiving as output a $(3/4)$-LLL-reduced basis B'. Third, it runs Babai's algorithm [Bab86] on $B', \boldsymbol{t}$, receiving as output $\boldsymbol{v} \in \mathcal{L}(B)$ satisfying $\|\boldsymbol{v} - \boldsymbol{t}\| \leq \gamma \cdot \mathrm{dist}(\boldsymbol{t}, \mathcal{L}(B))$ for $\gamma = \sqrt{n+1} \cdot 2^{(n-1)/2}$. The value

*Oregon State University, huck.bennett@oregonstate.edu.

[1] The *density* of a Subset Sum instance is defined as $n/(\max_{i\in[n]} \log a_i)$, which in terms of knapsack cryptography is roughly the ratio of the number of bits in the message $\boldsymbol{x} \in \{0,1\}^n$ to the number of bits in the ciphertext $s = \langle \boldsymbol{x}, \boldsymbol{a} \rangle$. Theorem 2 shows how to break random Subset Sum instances with density $\approx 2/n$.

of γ comes from the approximation factor guaranteed by Babai's algorithm together with the fact that B' is (3/4)-LLL-reduced. Finally, it outputs $\boldsymbol{y} := (v_2, \ldots, v_{n+1})^T \in \mathbb{Z}^n$, where we note that $B\boldsymbol{y} = \boldsymbol{v}$.

Because the LLL algorithm and Babai's algorithm run in polynomial time, it is clear that the algorithm is efficient. It remains to show correctness, i.e., that $\boldsymbol{y} = \boldsymbol{x}$ with overwhelming probability. First, because $\boldsymbol{x} \in \{0,1\}^n$ and $s = \langle \boldsymbol{x}, \boldsymbol{a} \rangle$, we have that

$$\operatorname{dist}(\boldsymbol{t}, \mathcal{L}(B)) \le \|B\boldsymbol{x} - \boldsymbol{t}\| = \|(0, x_1 - 1/2, x_2 - 1/2, \ldots, x_n - 1/2)^T\| = \sqrt{n}/2 \ . \tag{2}$$

Therefore, the vector $\boldsymbol{v}$ returned by Babai's algorithm must satisfy $\|\boldsymbol{v} - \boldsymbol{t}\| \le (\sqrt{n+1} \cdot 2^{(n-1)/2}) \cdot \sqrt{n}/2 < \beta$. We will show that with overwhelming probability over the choice of $\boldsymbol{a}$ no vector $\boldsymbol{v} \in \mathcal{L}(B)$ except for $\boldsymbol{v} = B\boldsymbol{x}$ satisfies this condition. For this, we make two observations about the coefficient vector $\boldsymbol{y} \in \mathbb{Z}^n$ of a lattice vector $B\boldsymbol{y}$ satisfying $\|B\boldsymbol{y} - \boldsymbol{t}\| < \beta$. First, it must hold that $\langle \boldsymbol{a}, \boldsymbol{y} \rangle = s$, since otherwise the first coordinate of $B\boldsymbol{y} - \boldsymbol{t}$ would have magnitude at least β. Second, it must hold that $\|\boldsymbol{y}\|_\infty \le \beta + 1$, since otherwise one of the last n coordinates of $B\boldsymbol{y} - \boldsymbol{t}$ would have magnitude at least $\beta + 1/2 > \beta$.

We first upper bound the probability that $\langle \boldsymbol{a}, \boldsymbol{y} \rangle = s$ for a fixed vector $\boldsymbol{y} \in \mathbb{Z}^n \setminus \{\boldsymbol{x}\}$. Assume without loss of generality that $y_1 \ne x_1$. Then

$$\Pr_{a_1,\ldots,a_n}\Big[\sum_{i=1}^n y_i a_i = s\Big] = \Pr_{a_1,\ldots,a_n}\Big[\sum_{i=1}^n y_i a_i = \sum_{i=1}^n x_i a_i\Big] = \Pr_{a_1}\Big[\Big(\sum_{i=2}^n (y_i - x_i) a_i\Big)/(y_1 - x_1) = -a_1\Big] \le 1/N \ , \tag{3}$$

where the inequality follows because a_1 is sampled uniformly at random from $\{1, \ldots, N\}$.

Second, we note that there are $(2\beta+3)^n$ vectors $\boldsymbol{y} \in \mathbb{Z}^n$ with $\|\boldsymbol{y}\|_\infty \le \beta + 1$. Taking a union bound over these vectors and using Equation (3), we have that the probability that there exists $\boldsymbol{y} \in \mathbb{Z}^n \setminus \{\boldsymbol{x}\}$ satisfying $\|B\boldsymbol{y} - \boldsymbol{t}\| < \beta$ is at most

$$(2\beta+3)^n/N \le 2^{n^2/2 + o(n^2)}/2^{(1/2+\varepsilon)n^2} \le 2^{-(\varepsilon - o(1))n^2} \ .$$

It follows that the algorithm must output $\boldsymbol{x}$ with probability at least $1 - 2^{-(\varepsilon - o(1))n^2}$, as needed. □

We note that the basis and target defined in Equation (1) are essentially identical to the ones output in the standard reduction from Subset Sum to GapCVP (the exact, decision version of CVP). Indeed, when invoked with worst-case instances of Subset Sum, essentially the same reduction that we used above implies that GapCVP is NP-hard. (When using this reduction to prove NP-hardness of GapCVP, multiplying the top row of B and first coordinate of $\boldsymbol{t}$ by the scaling factor β is unnecessary, but does not hurt anything.)

References

[Bab86] László Babai. On Lovász' lattice reduction and the nearest lattice point problem. *Comb.*, 6(1):1–13, 1986.

[Fri86] Alan M. Frieze. On the lagarias-odlyzko algorithm for the subset sum problem. *SIAM J. Comput.*, 15(2):536–539, 1986.

[LLL82] A. K. Lenstra, H. W. Lenstra, and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):515–534, 1982.

[LO85] J. C. Lagarias and Andrew M. Odlyzko. Solving low-density subset sum problems. *J. ACM*, 32(1):229–246, 1985.

[MH78] Ralph C. Merkle and Martin E. Hellman. Hiding information and signatures in trapdoor knapsacks. *IEEE Trans. Inf. Theory*, 24(5):525–530, 1978.

[Pei15] Chris Peikert. Cryptanalysis of knapsack crypto, 2015. Lecture notes. Available at https://web.eecs.umich.edu/~cpeikert/lic15/lec05.pdf.