**2.4.3.1 Finding a Short Basis**

The short basis algorithm involves finding a short basis for a lattice with respect to = + . To put it simply, “the goal of lattice basis reduction is to find a basis with short, nearly orthogonal (perpendicular) vectors when given an integer lattice basis as input” (Wikipedia). The input for the algorithm includes two vectors ​ and , which form an integral lattice basis, along with a positive integer q. The output should be two vectors, and , that form a short basis for the same lattice. The input must also meet these requirements: ≤ . Now that we know what to input, let’s talk about the process. The process starts by initializing and swapping the basis vectors to maintain the condition ≤ ). The algorithm then enters a loop, where it computes a value r that is rounded by dividing . The goal of the loop is to reduce the length of the vectors. The loop continues It updates the gamma vector by subtracting r from , and if the norm of gamma is smaller than , the vectors are updated. Finally, the loop terminates, and the short basis vectors and are returned.

**2.4.3.2 Finding the Closest Vector**

After computing a reduced basis, the next step is to find a vector in the lattice that is closest to the target vector t. This is accomplished using Babai’s nearest plane algorithm. The nearest plane algorithm was developed by L. Babai in 1986, and it “obtains a 2(2/√3) ^n approximation ratio, where n is the rank of the lattice” (Regev). In other words, the algorithm projects the target vector onto the reduced basis, and then adjusts the result to produce a vector c that is close to t with respect to the norm .

**2.4.3.3 Enumerating Close Vectors**

The final step is to enumerate short vectors in the lattice that are close to the target vector t, using the Fincke-Pohst algorithm. In other words, “the running time of enumeration algorithms greatly depends on the quality of the input lattice basis. So, suitably preprocessing the input lattice using a basis reduction algorithm is an essential part of lattice enumeration methods” (Lattice Cryptography). Furthermore, the algorithm takes as input a lattice L, a target vector t, and an initial close vector c. It then iterates through possible values for the coordinates x and y, generating vectors that are within bound B from the target vector. By updating both x and y within nested loops, the algorithm yields vectors that satisfy the distance condition. This process continues until either the specified number of tries is reached, or all close vectors are found.

**2.4.4.1 Basic Quaternion Arithmetic**

Quaternions are generalizations of complex numbers, which are represented by the expression: Alpha = a + bi +cj + dk / r. The basic arithmetic operations like addition and multiplication are computed by reducing common denominators where necessary. Specifically, multiplication follows the axioms: i^2 = -1, j^2 = -p, ij = -ji = k. We also have the conjugate operation where alpha prime = a- bi - cj -dk / r. There’s also the reduced trace that simplifies to: tr(alpha) = 2a / r. Finally, we have the reduced norm: nrd(alpha) = a^2 + b^2 + p (c^2 + d^2) / r^2.

* + - 1. **Lattices**

A lattice is a collection of quaternions that form a grid-like structure. Additionally, lattices are defined by a basis of quaternions that are represented as columns of a matrix. You can also perform operations on lattices such as:

Equality: Whether two transactions are similar by comparing their HNF.

Union and Intersection: The union of two lattices is formed by merging their initial matrices into HNF. The segmentation of the dual network is obtained by computing the HNF.

Multiplication: In lattice multiplication, the base matrices are multiplied by the corresponding quaternary algebra generators.

Containment: This function ensures that an element is contained in the network by solving a system of linear equations.

Index: The index of one network in another is the mean value of their basis matrices.

Right Transporter: The right transporter of a lattice is a set of objects that, when connected by a grid, still belong to another grid. It is one of the most complex operations on networks. (SQI Sign)

**2.4.5 Quaternion orders and ideals**

Quaternaries play an important role in algebra, and their structure includes some small orders and ideals. Orders are discrete relations in quaternary algebras, while ideals are subsets of these relations. Operations on quaternary orders and ideals underlie many algebraic programs. For example, we have a benchmark for our model. Here, the standard deviation of ideal I is defined as the most common divisor of the values ​​of its components. This is important because it provides a measure of the size.

**2.4.5.1 Basic Operations on Ideals**

Now that we know about the importance of orders and ideals, it is also important to note that various operations can be performed on them. Two ideals are considered isomorphic if there exists a quaternion that scales one ideal into the other. A key operation is finding the connecting ideal between two orders and ​. This connecting ideal is defined by the intersection of the two orders and helps bridge the relationship between them. When two ideals are multiplied together, the result is another ideal, which retains the properties of the original ideals.

**2.4.5.2 Finding Equivalent Ideals of Small Norm**

Sometimes, it’s necessary to find an equivalent ideal with a smaller norm, particularly when working with large ideals. This process involves random sampling to reduce the norm, starting with a reduced basis for the ideal. The algorithm iterates through smaller and smaller norms until it finds a small norm. This is useful in reducing complexity when handling large ideals. By finding ideals with smaller norms, calculations become more efficient, which is crucial in many applications involving quaternion algebras.

**Bibliography**

Jorge Chavez-Saab, Maria Corte-Real Santos, Luca DeFeo, Jonathan Komada Eriksen, Basil Hess, David Kohel, Antonin Leroux, Patrick Longa, Michael Meyer, Lorenz Panny, Sikhar Patranabis, Christophe Petit, Francisco Rodríguez Henríquez, Sina Schaeffler, and Benjamin Wesolowski. SQISIGN (June 2023)

“Lattice Cryptography.” *Enum*, cseweb.ucsd.edu/~daniele/LatticeLinks/Enum.html. Accessed 23 Sept. 2024.

“Lattice Reduction.” *Wikipedia*, Wikimedia Foundation, 22 Jan. 2024, en.wikipedia.org/wiki/Lattice\_reduction.

Regev, Oded. *Lecture 3 CVP Algorithm 1 the Nearest Plane Algorithm*, cims.nyu.edu/~regev/teaching/lattices\_fall\_2004/ln/cvp.pdf. Accessed 23 Sept. 2024.