C

The Root Lattices

Define $\mathbf{e}_i^L$ to be a vector in L dimensions whose ith component is 1 and all other components are 0. Some of the root systems that are used in lattice vector quantization are given as follows:

$$
\begin{array}{lll}
D_L & \pm\mathbf{e}_i^L \pm \mathbf{e}_j^L, & i \neq j,\ i, j = 1, 2, \ldots, L \\
A_L & \pm(\mathbf{e}_i^{L+1} - \mathbf{e}_j^{L+1}), & i \neq j,\ i, j = 1, 2, \ldots, L \\
E_L & \pm\mathbf{e}_i^L \pm \mathbf{e}_j^L, & i \neq j,\ i, j = 1, 2, \ldots, L-1, \\
 & \frac{1}{2}(\pm\mathbf{e}_1 \pm \mathbf{e}_2 \cdots \pm \mathbf{e}_{L-1} \pm \sqrt{2 - \frac{(L-1)}{4}}\mathbf{e}_L) & L = 6, 7, 8
\end{array}
$$

Let us look at each of these definitions a bit closer and see how they can be used to generate lattices.

D_L Let us start with the D_L lattice. For $L = 2$, the four roots of the D_2 algebra are $\mathbf{e}_1^2 + \mathbf{e}_2^2$, $\mathbf{e}_1^2 - \mathbf{e}_2^2$, $-\mathbf{e}_1^2 + \mathbf{e}_2^2$, and $-\mathbf{e}_1^2 - \mathbf{e}_2^2$, or (1, 1), (1, −1), (−1, 1), and (−1, −1). We can pick any two independent vectors from among these four to form the basis set for the D_2 lattice. Suppose we picked (1, 1) and (1, −1). Then any integral combination of these vectors is a lattice point. The resulting lattice is shown in Figure 10.24 in Chapter 10. Notice that the sum of the coordinates are all even numbers. This makes finding the closest lattice point to an input a relatively simple exercise.

A_L The roots of the A_L lattices are described using $L+1$-dimensional vectors. However, if we select any L independent vectors from this set, we will find that the points that are generated all lie in an L-dimensional slice of the $L+1$-dimensional space. This can be seen from Figure C.1.

We can obtain an L-dimensional basis set from this using a simple algorithm described in [139]. In two dimensions, this results in the generation of the vectors $(1, 0)$ and $(-\frac{1}{2}, \frac{\sqrt{3}}{2})$. The resulting lattice is shown in Figure 10.25 in Chapter 10. To find the closest point to the A_L lattice, we use the fact that in the embedding of the lattice in $L+1$ dimensions, the sum of the coordinates is always zero. The exact procedure can be found in [141, 140].

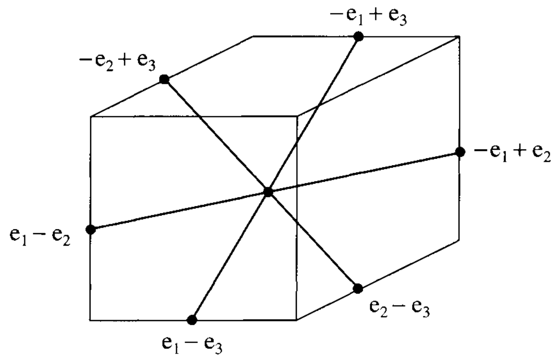

FIGURE C. 1 **The A_2 roots embedded in three dimensions.**

E_L As we can see from the definition, the E_L lattices go up to a maximum dimension of 8. Each of these lattices can be written as unions of the A_L and D_L lattices and their translated version. For example, the E_8 lattice is the union of the D_8 lattice and the D_8 lattice translated by the vector $(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2})$. Therefore, to find the closest E_8 point to an input $\mathbf{x}$, we find the closest point of D_8 to $\mathbf{x}$, and the closest point of D_8 to $\mathbf{x} - (\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2})$, and pick the one that is closest to $\mathbf{x}$.

There are several advantages to using lattices as vector quantizers. There is no need to store the codebook, and finding the closest lattice point to a given input is a rather simple operation. However, the quantizer codebook is only a subset of the lattice. How do we know when we have wandered out of this subset, and what do we do about it? Furthermore, how do we generate a binary codeword for each of the lattice points that lie within the boundary? The first problem is easy to solve. Earlier we discussed the selection of a boundary to reduce the effect of the overload error. We can check the location of the lattice point to see if it is within this boundary. If not, we are outside the subset. The other questions are more difficult to resolve. Conway and Sloane [142] have developed a technique that functions by first defining the boundary as one of the quantization regions (expanded many times) of the root lattices. The technique is not very complicated, but it takes some time to set up, so we will not describe it here (see [142] for details).

We have given a sketchy description of lattice quantizers. For a more detailed tutorial review, see [140]. A more theoretical review and overview can be found in [262].