Contents

Chapter 1

Classes

1.1 lattice – Lattice

- **Classes**
  - **Lattice**
  - **LatticeElement**
- **Functions**
  - **LLL**

1.1.1 Lattice – lattice

Initialize (Constructor)

Lattice(`basis`: **RingSquareMatrix**, `quadraticForm`: **RingSquareMatrix**)
→ ***Lattice***

Create Lattice object.

Attribute

basis : The basis of `self` lattice.

quadraticForm : The quadratic form corresponding the inner product.

Methods

1.1.1.1 createElement – create element

createElement(self, compo: ***list***) → **LatticeElement**

Create the element which has coefficients with given `compo`.

1.1.1.2 bilinearForm – bilinear form

bilinearForm(self, v_1: **Vector**, v_2: **Vector**) → ***integer***

Return the inner product of v_1 and v_2 with **quadraticForm**.

1.1.1.3 isCyclic – Check whether cyclic lattice or not

isCyclic(self) → ***bool***

Check whether `self` lattice is a cyclic lattice or not.

1.1.1.4 isIdeal – Check whether ideal lattice or not

signature(self) → ***bool***

Check whether `self` lattice is an ideal lattice or not.

1.1.2 LatticeElement – element of lattice

Initialize (Constructor)

LatticeElement(`lattice`**: Lattice**, `compo`**: *list*,)** $\rightarrow$ ***LatticeElement***

Create LatticeElement object.

Elements of lattices are represented as linear combinations of basis. The class inherits **Matrix**. Then, intances are regarded as $n \times 1$ matrix whose coefficients consist of `compo`, where n is the dimension of lattice.

`lattice` is an instance of Lattice object. `compo` is coeeficients list of basis.

Attribute

lattice : the lattice which includes `self`

Methods

1.1.2.1 getLattice – Find lattice belongs to

getLattice(`self`) → Lattice

Obtain the Lattice object corresponding to `self`.

1.1.3 LLL(function) – LLL reduction

LLL(`M`: RingSquareMatrix) → `L`: RingSquareMatrix, `T`: RingSquareMatrix

Return LLL-reduced basis for the given basis `M`.

The output `L` is the LLL-reduced basis. `T` is the transportation matrix from the original basis to the LLL-reduced basis.

Examples

```
>>> M=mat.Matrix(3,3,[1,0,12,0,1,26,0,0,13]);
>>> lat.LLL(M);
([1, 0, 0]+[0, 1, 0]+[0, 0, 13], [1L, 0L, -12L]+[0L, 1L, -26L]+[0L, 0L, 1L])
```

Bibliography