Algorithms in lattice-based cryptohraphy

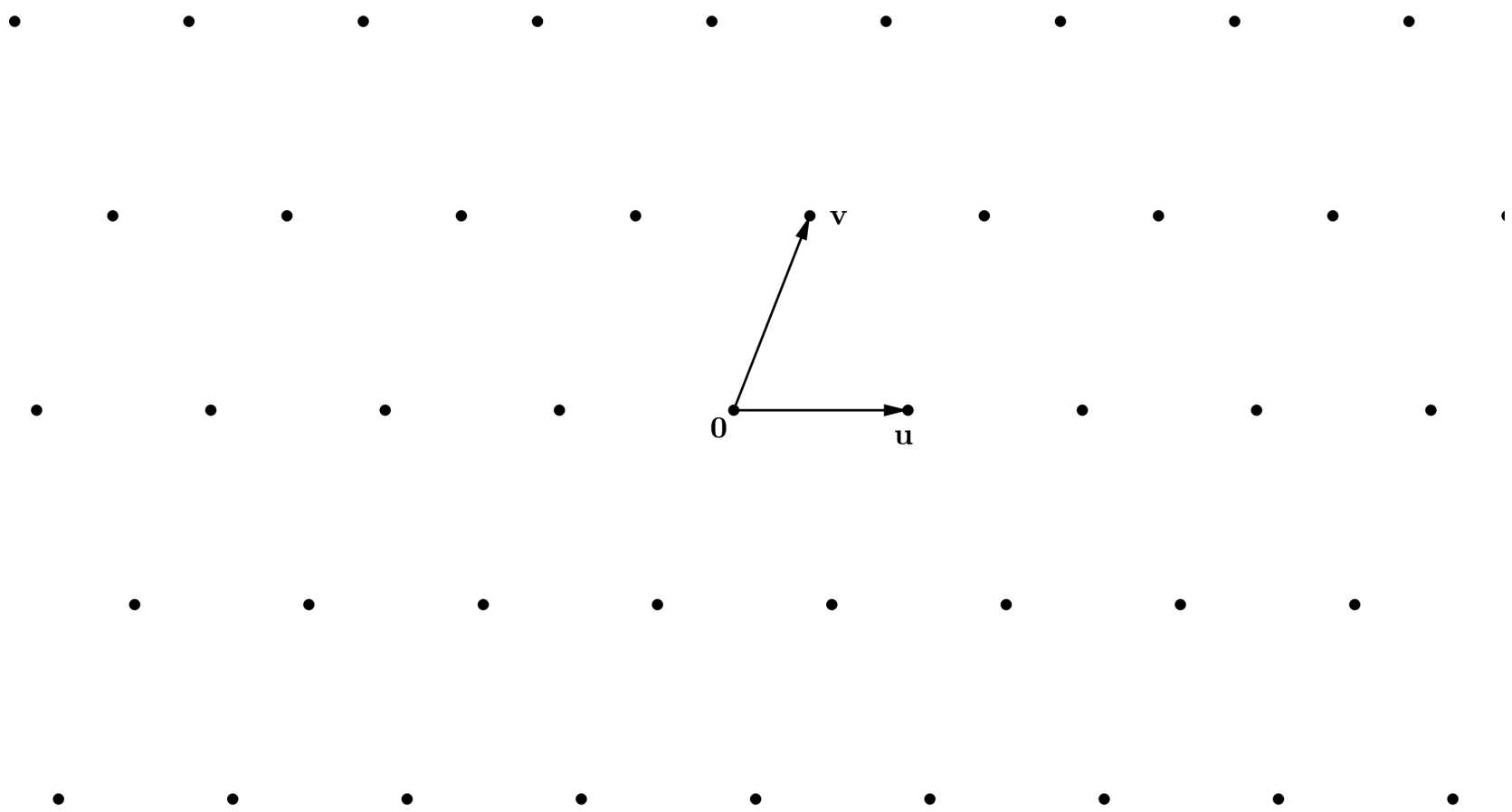
June 5, 2025

Overview

- Lattices
- Good and bad lattice bases
- Algorithmic problems
- Relevance in cryptography

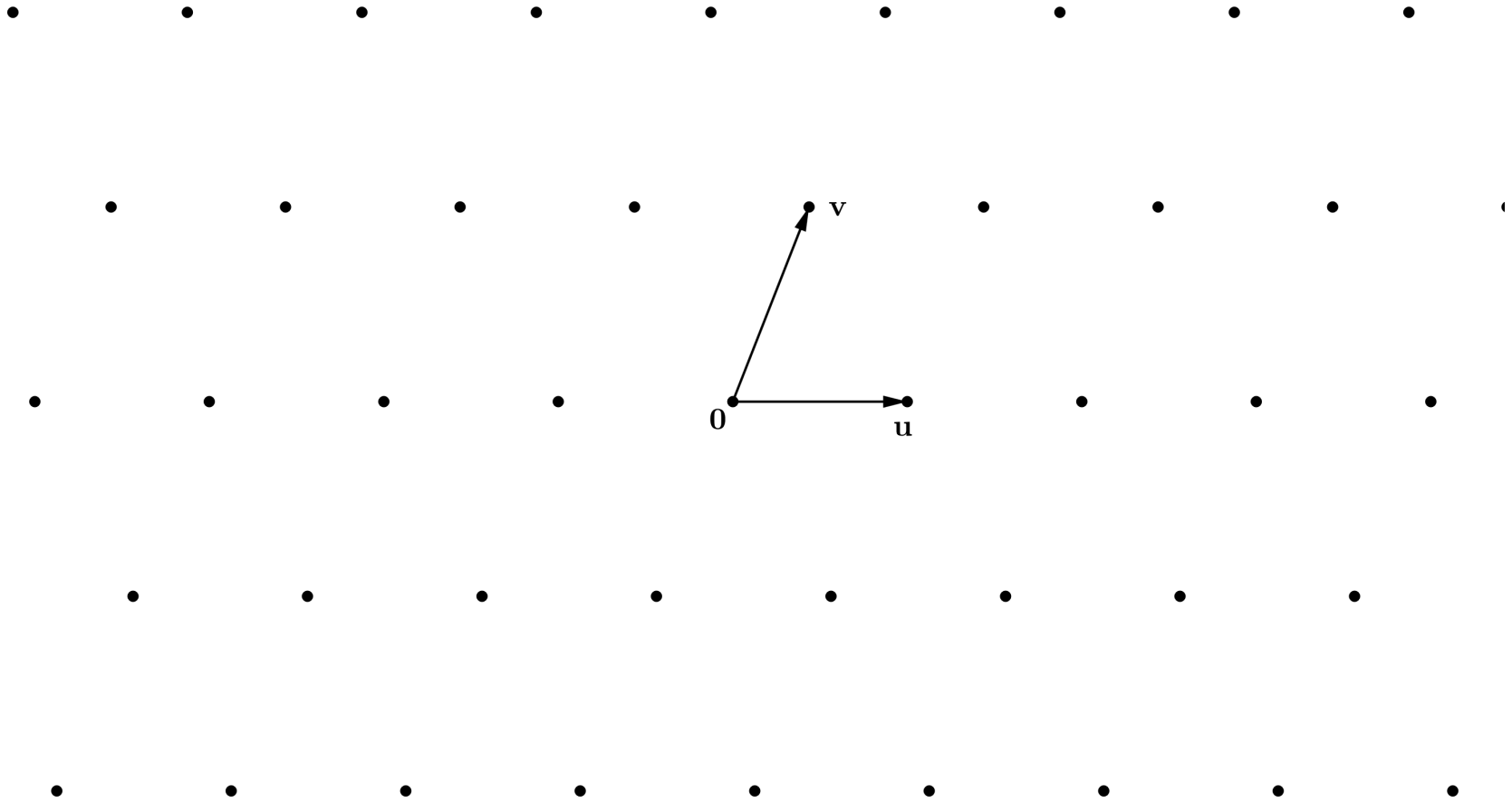
Lattices

- Lattice: $L = \{s_1 b_1 + \ldots + s_n b_n \mid s_i \in \mathbb{Z}\}$ where $b_1, \ldots, b_n$ is a basis of $\mathbb{R}^n$.



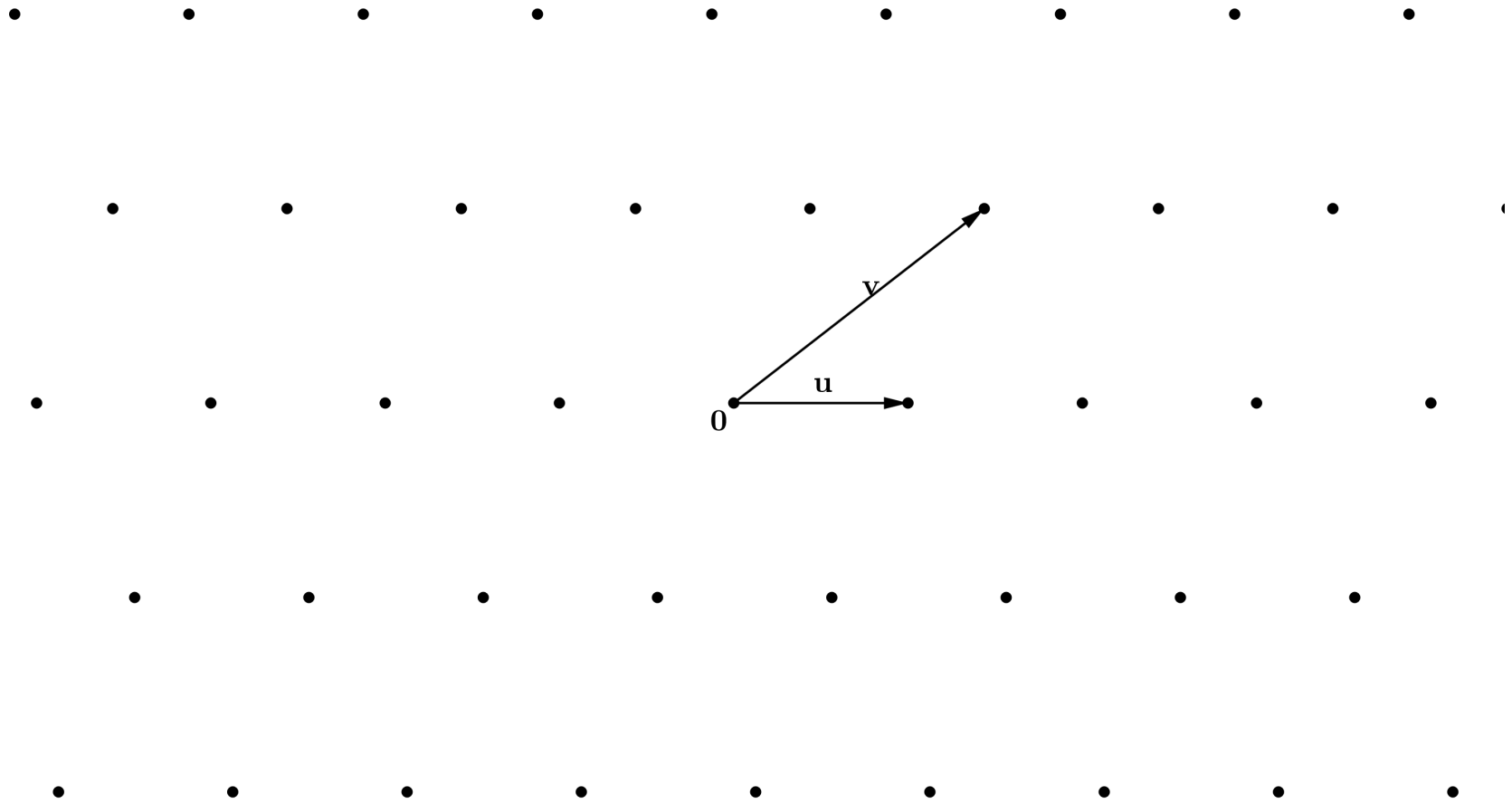
Lattices

- Infinitely many bases



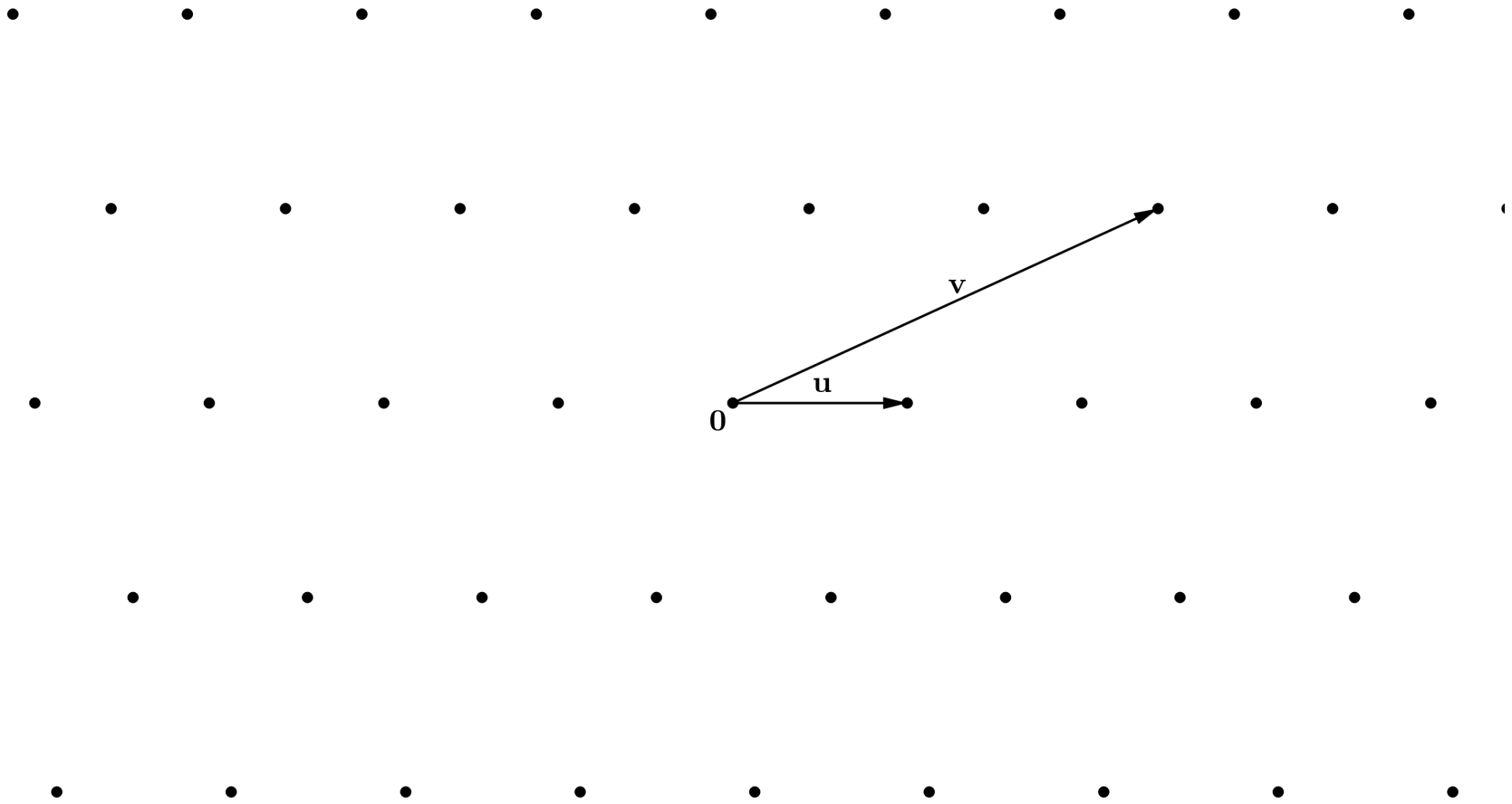
Lattices

- Infinitely many bases



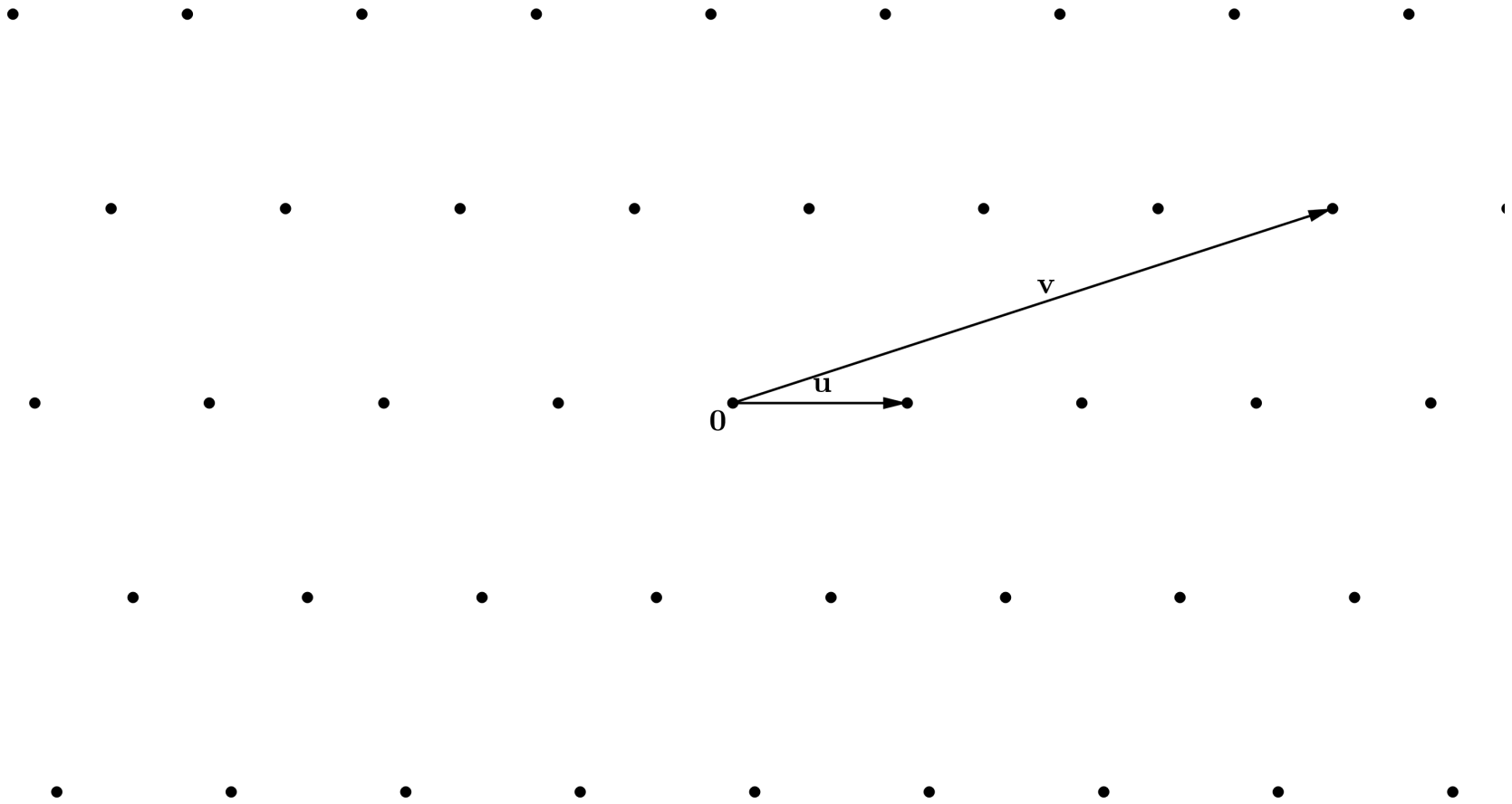
Lattices

- Infinitely many bases



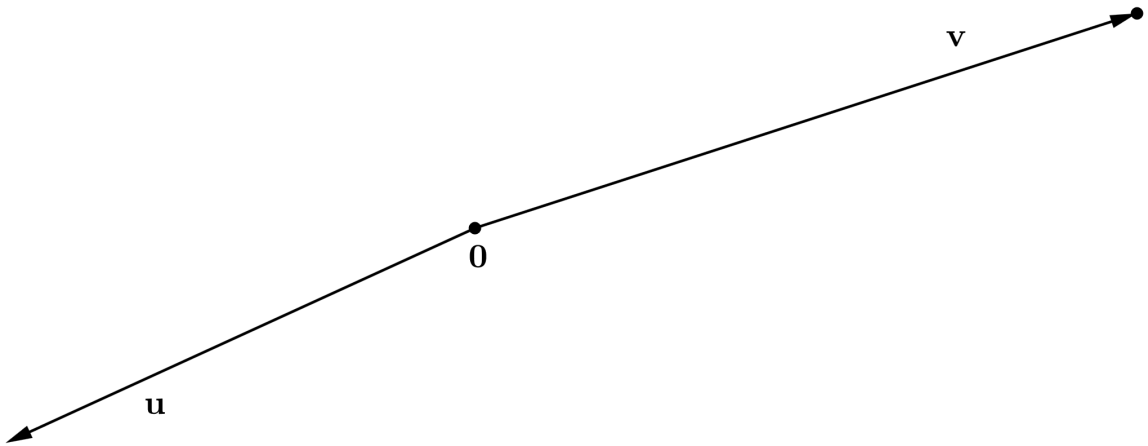
Lattices

- Infinitely many bases



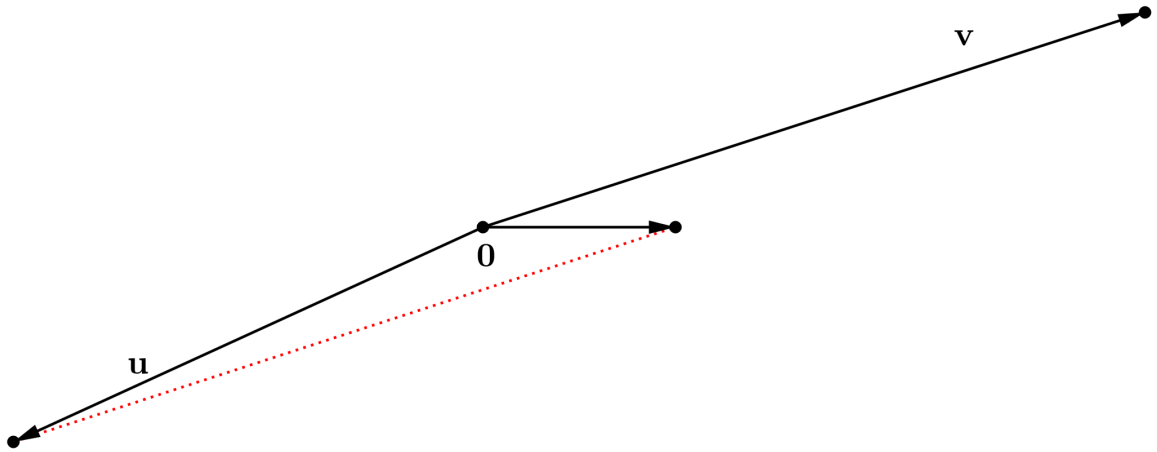
Bad and good lattice bases

- A bad base



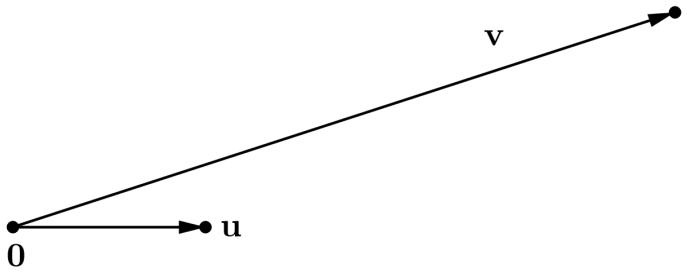
Bad and good lattice bases

- $u \leftarrow u + v$



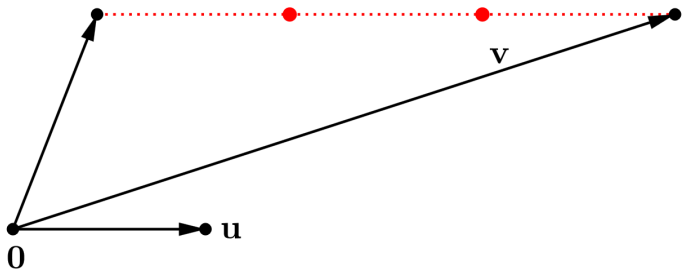
Bad and good lattice bases

- $u \leftarrow u + v$



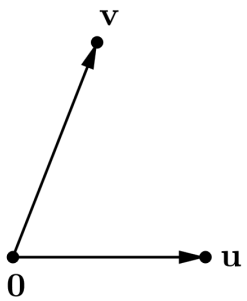
Bad and good lattice bases

- $v \leftarrow v - 3u$

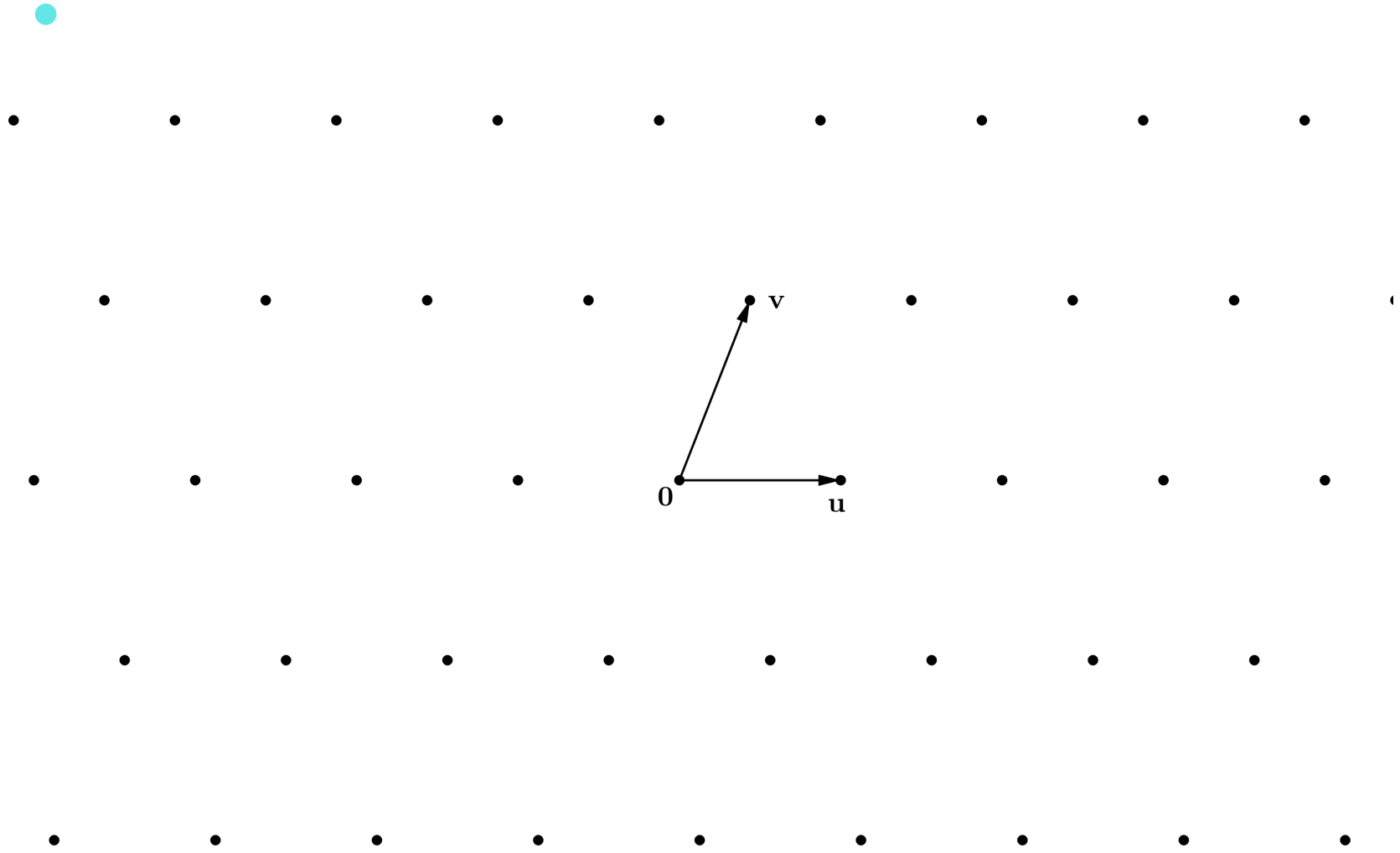


Bad and good lattice bases

- $v \leftarrow v - 3u$

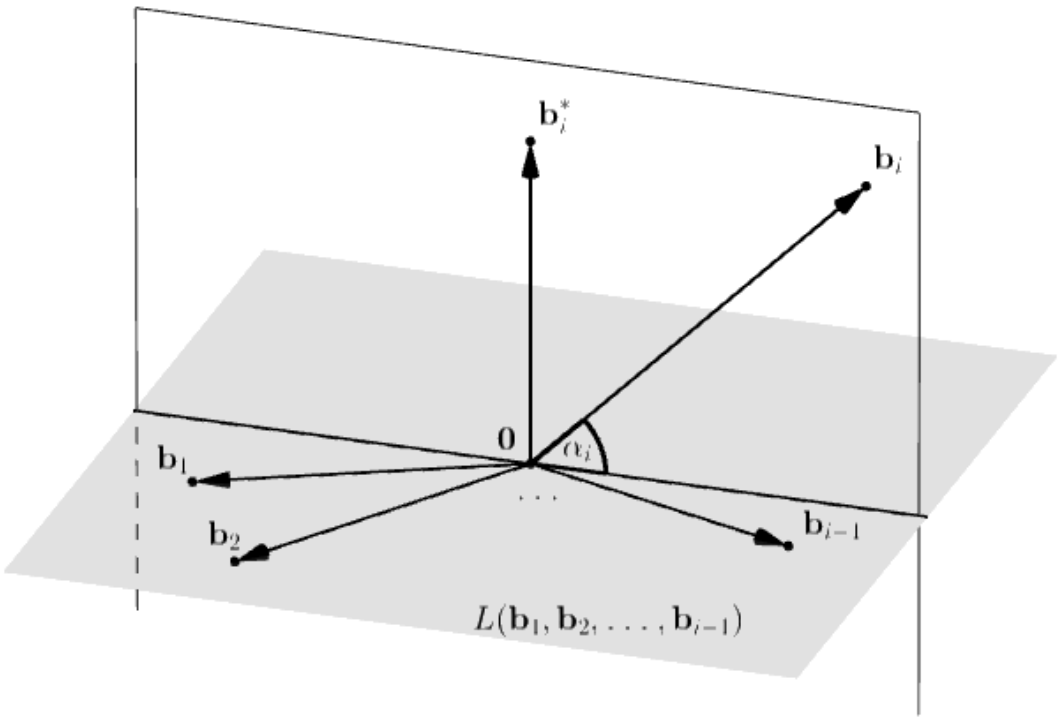


Bad and good lattice bases



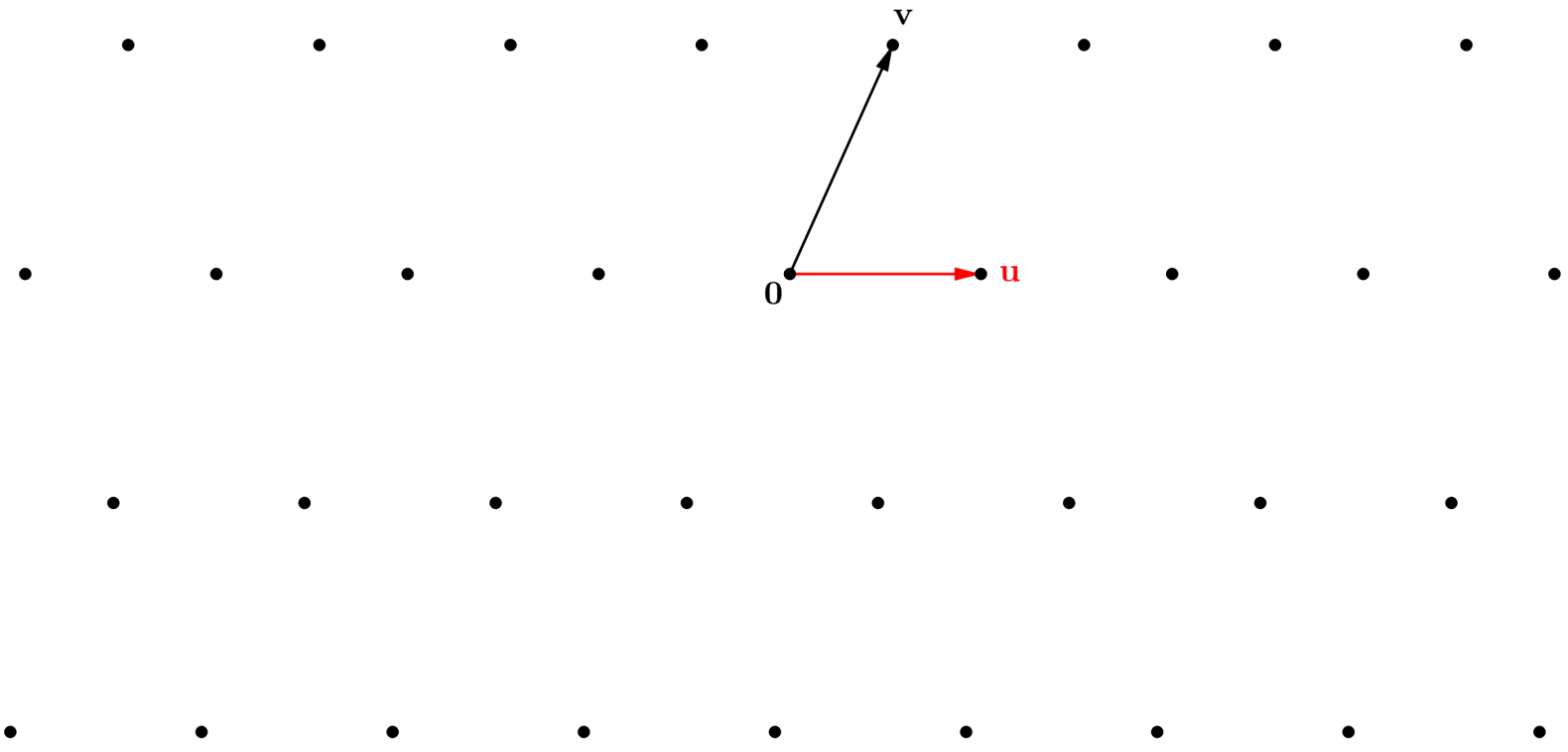
The orthogonality-defect

- A constant describeing how orthogonal a basis is
- $\delta(\mathsf{b}_1, \ldots, \mathsf{b}_n) = \prod_{i=1}^{n} \frac{\|\mathsf{b}_i\|}{\|\mathsf{b}_i^*\|}$



SVP

- Given a basis $b_1, \ldots, b_n$ of the lattice L, find $x \in L \setminus \{0\}$ s.t. $\|x\|$ is minimal



SVP (Shortest Vector Problem)

- Using enumeration SVP can be solved in $3^n\delta(\mathrm{b}_1, \ldots, \mathrm{b}_n)$ time and polynomial space.

SVP (Shortest Vector Problem)

- Using enumeration SVP can be solved in $3^n\delta(b_1, \ldots, b_n)$ time and polynomial space.
- There are lattices where

$$\min \delta(b_1, \ldots, b_n) \approx n^{n/2}$$

SVP (Shortest Vector Problem)

- Using enumeration SVP can be solved in $3^n\delta(b_1,\ldots,b_n)$ time and polynomial space.
- There are lattices where

$$\min \delta(b_1,\ldots,b_n) \approx n^{n/2}$$

- Running time $n^{O(n)}$, this is currently the best among polynomial space algorithms

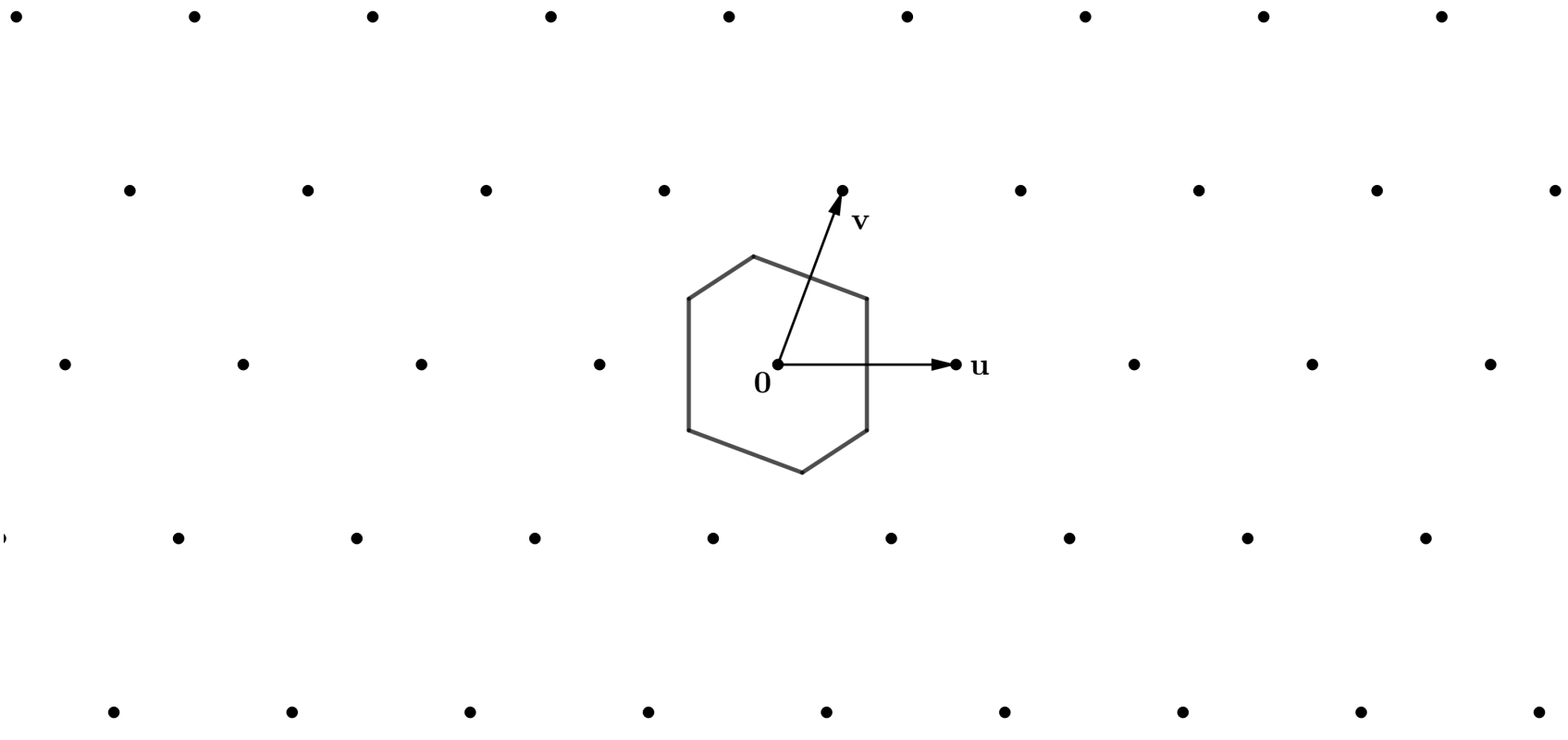
Open questions

- There are $2^{O(n)}$ time and space algorithms
- Hard open question: Is there $2^{O(n)}$ time and polynomial space algorithm?

The Voronoi-cell and the covering radius

- Voronoi cell of lattice L:

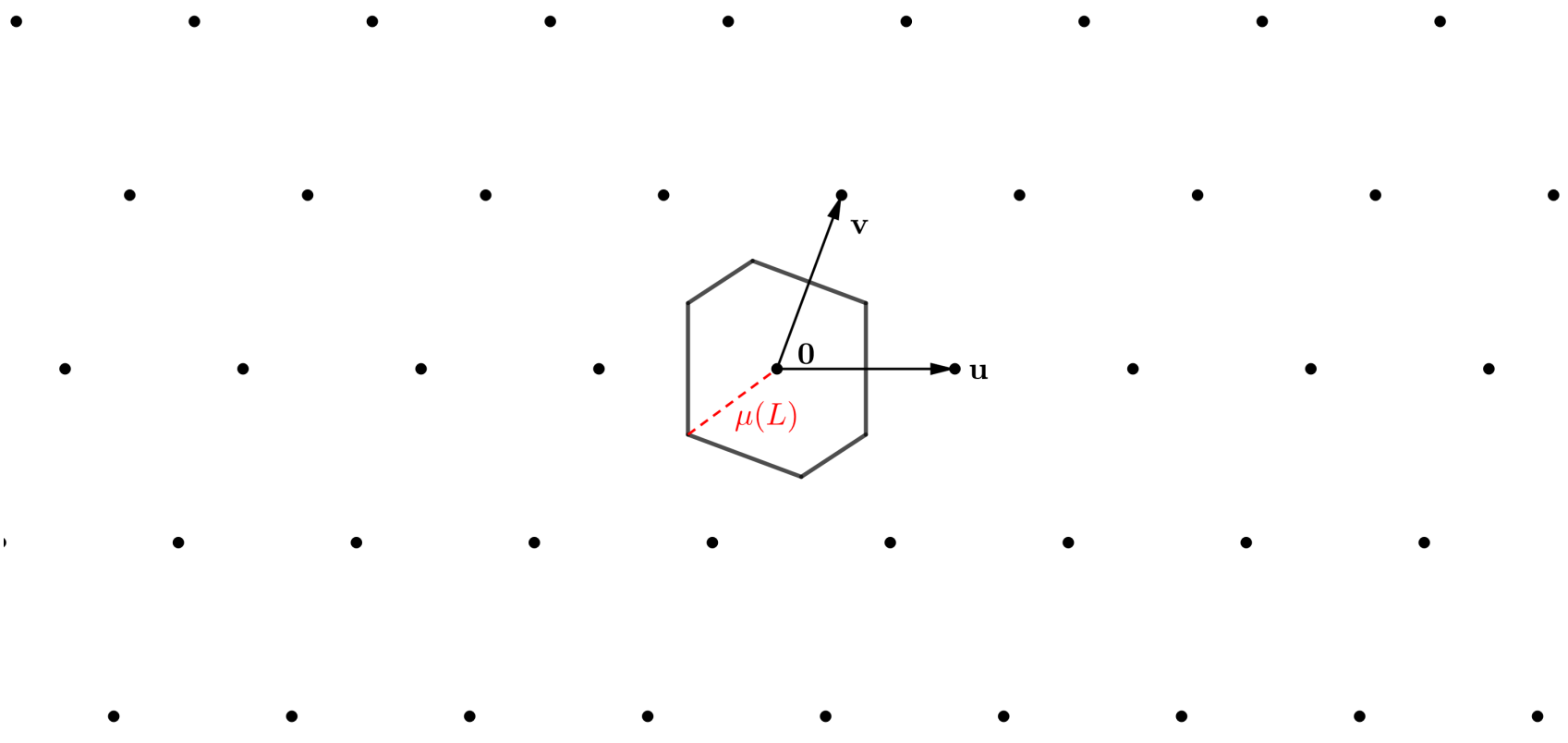
$$V(L) = \{x \in \mathbb{R}^n \mid \forall y \in L\colon d(x,0) \leq d(x,y)\}$$



The Voronoi-cell and the covering radius

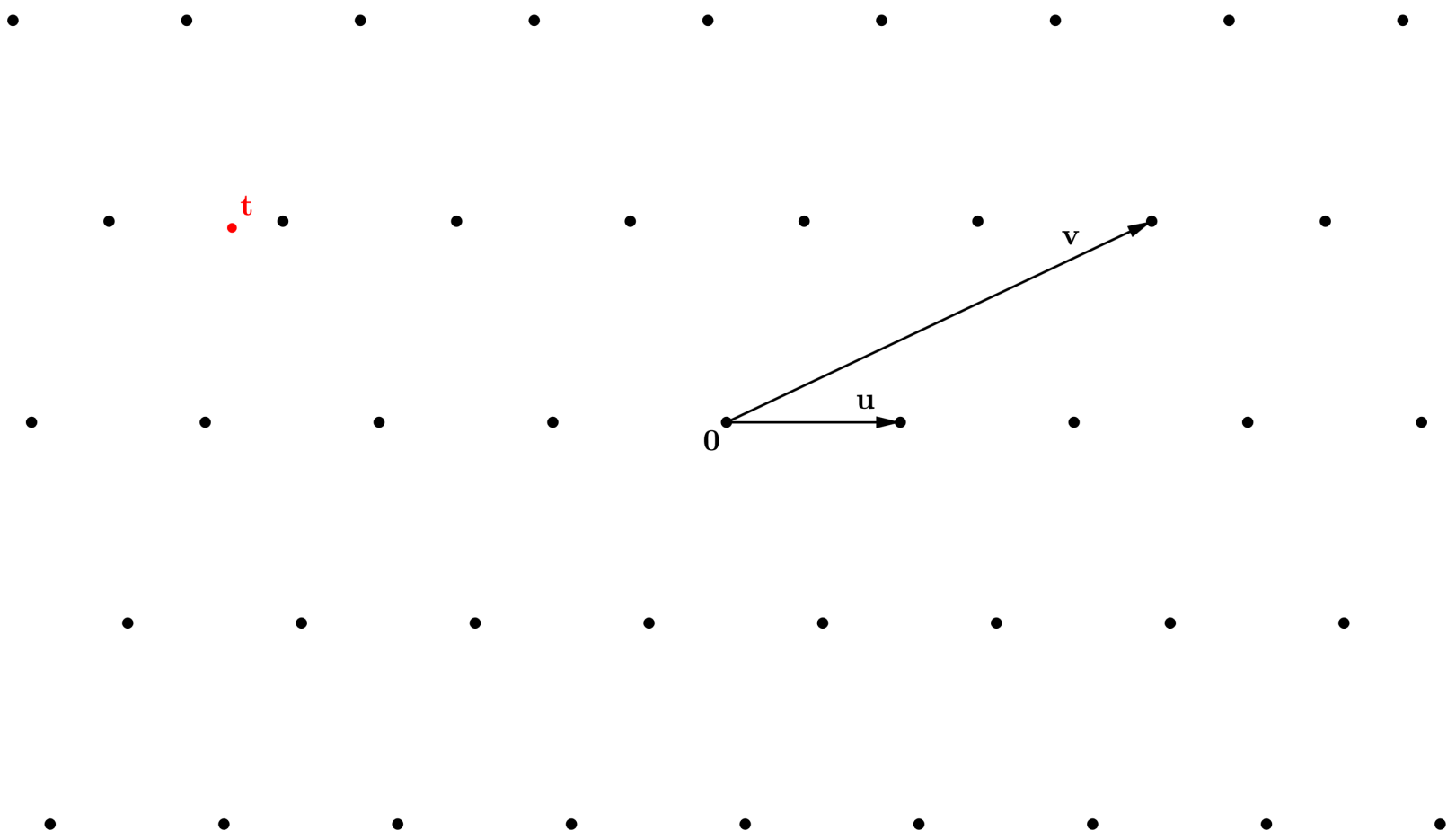
- Covering radius of L:

$$\mu(L) = \sup_{x \in V(L)} \|x\|$$



CVP (Closest Vector Problem)

- Given a basis $b_1, \ldots, b_n$ of a lattice L and a target vector $t \in \mathbb{R}^n$, find $x \in L$ closest to t.



An other constant

- An other way of measuring how good a basis is:

$$\mu(\mathsf{b}_1, \ldots, \mathsf{b}_n) = \prod_{i=1}^{n} \frac{2\mu(L_i)}{\|\mathsf{b}_i^*\|}$$

  where L_i is the lattice spanned by $\mathsf{b}_1, \ldots, \mathsf{b}_i$.
- CVP can be solved using enumeration in time $2^n \mu(\mathsf{b}_1, \ldots, \mathsf{b}_n)$
- By reduction on the dual lattice, one can achieve $\mu(\mathsf{b}_1, \ldots, \mathsf{b}_n) \leq n!$ (using heuristics one can get $\leq n^{n/2}$)

My observations

- One can use an approximation version of the dual basis reduction: this gives a faster running time for reduction, and slower running time for enumeration
- One can bound the running time of SVP using a slightly different constant:

$$\prod_{i=1}^{n-1} \frac{2\mu(L_i)}{\|\mathbf{b}_{i+1}^*\|}$$

which has a different geometric meaning

Importance in cryptography

- Average-case problems $\leftarrow$ worst-case hard problems
- (Assumed to be) safe against quantum attacks (Shor's quantum algorithm breaks RSA)
- Modern crypto builds on these and similar lattice problems