THE MATHEMATICS OF LATTICE-BASED CRYPTOGRAPHY

4. Lattices

Alfred Menezes
cryptography101.ca

Outline

1. Definition of a lattice

2. Characterization of the bases of a lattice

3. Successive minima

4. LLL lattice basis reduction algorithm

5. SVP

6. SIVP

Lattice definition

- **Definition**. A *lattice* L in $\mathbb{R}^n$ is the set of all integer linear combinations of m linearly independent vectors $B = \{v_1, v_2, \ldots, v_m\}$ in $\mathbb{R}^n$ (and where $m \leq n$). The set B is called a *basis* of L, and we write $L = L(B)$. The *dimension* of L is n, and the *rank* of L is m.

- **Notes**:

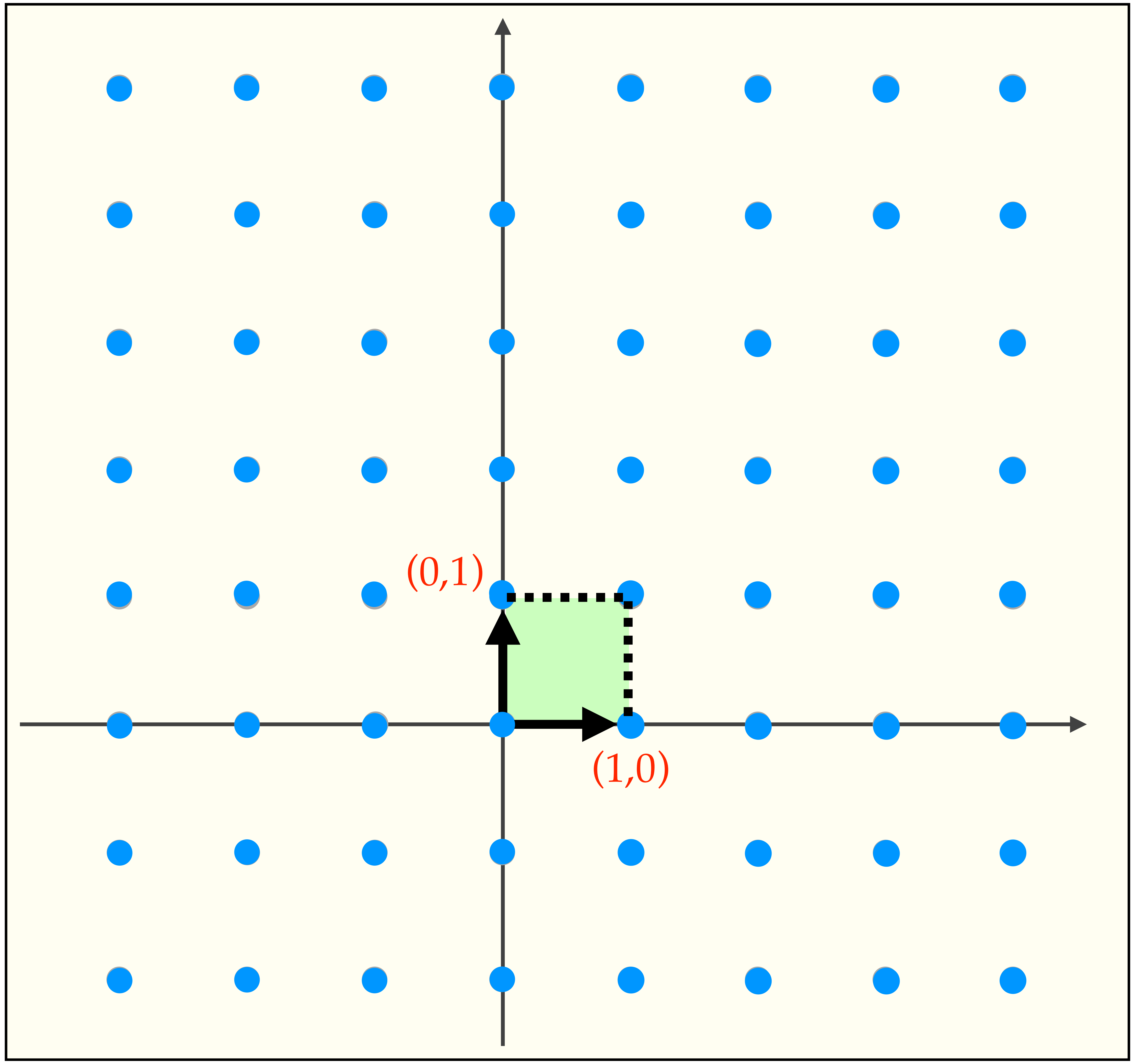
  1. We will henceforth assume that the basis vectors $v_1, v_2, \ldots, v_m$ are in $\mathbb{Z}^n$.
  2. Thus, $L = \{x_1 v_1 + x_2 v_2 + \cdots + x_m v_m \ : \ x_1, x_2, \ldots, x_m \in \mathbb{Z}\} \subseteq \mathbb{Z}^n$.
     L is called an *integer lattice*.
  3. Let B be the $n \times m$ matrix whose columns are the basis vectors $v_1, \ldots, v_m$,
     so $B = \begin{bmatrix} | & | & \cdots & | \\ v_1 & v_2 & \cdots & v_m \\ | & | & \cdots & | \end{bmatrix}$. Then $L = \{Bx : x \in \mathbb{Z}^m\}$.

Full-rank lattices

- **Definition**. A *full-rank lattice* L in $\mathbb{R}^n$ is a lattice in $\mathbb{R}^n$ of rank n.
- **Definition**. Let L and L' be lattices in $\mathbb{R}^n$.
  Then L' is a *sublattice* of L if $L' \subseteq L$.
- Henceforth, unless otherwise stated, all lattices and sublattices will be full-rank (and integer).
- Note that a basis $B = \{v_1, v_2, \ldots, v_n\}$ for a full-rank lattice in $\mathbb{R}^n$ is also a basis for the vector space $\mathbb{R}^n$.

Lattice: Example 1

- Let $n = 2$ and $B_1 = \{(1,0), (0,1)\}$.
- Then $L_1 = L(B_1) = \{B_1 x : x \in \mathbb{Z}^2\}$, where $B_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.
- Thus, $L_1 = \mathbb{Z}^n$.
- **Fundamental parallelepiped**: $P(B_1) = \{a_1(1,0) + a_2(0,1) : a_1, a_2 \in [0,1)\}$.



Fundamental parallelepiped

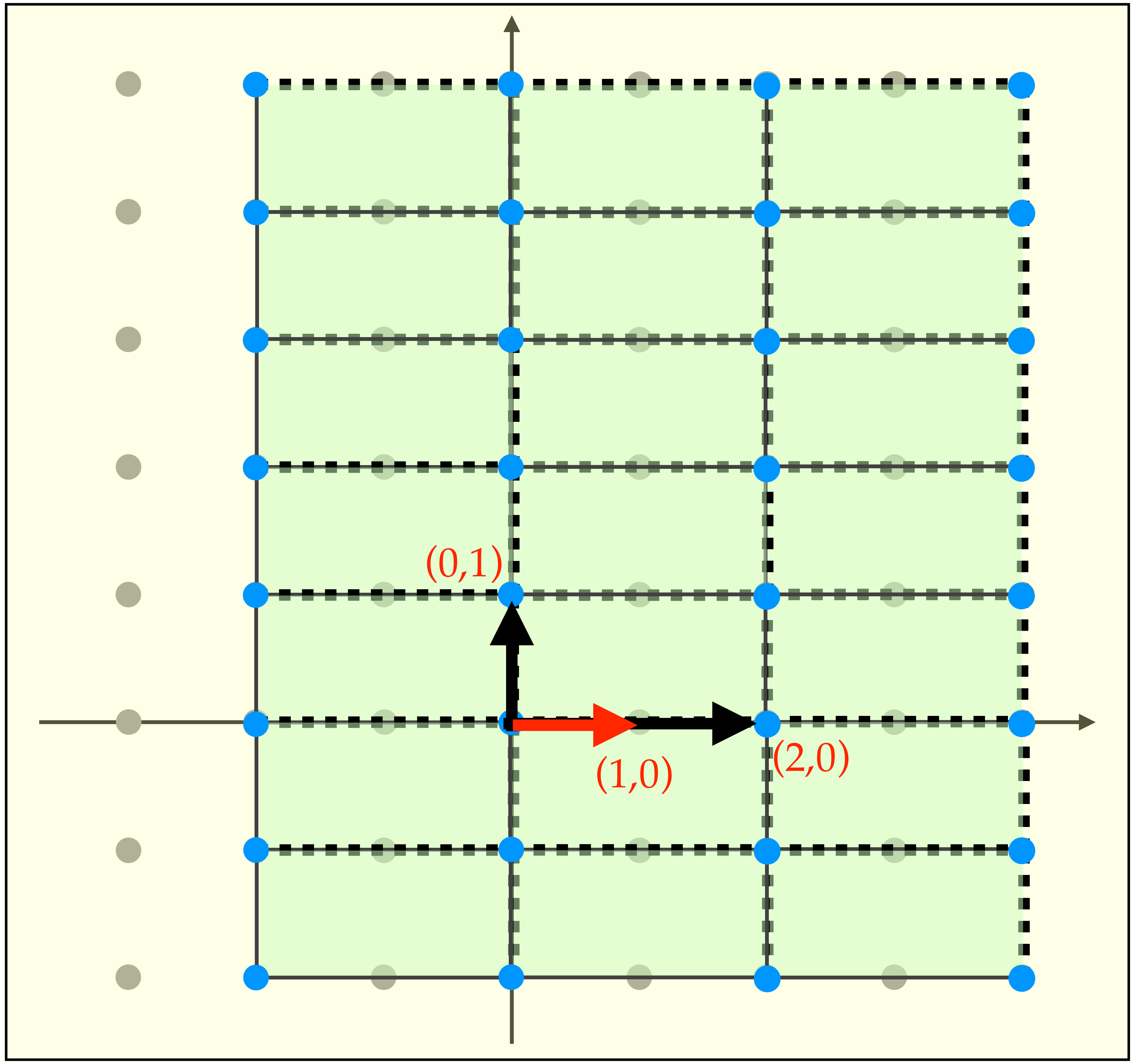
- **Definition**. Let $L = L(B)$ be a lattice in $\mathbb{R}^n$, where $B = \{v_1, v_2, \ldots, v_n\}$.
  The *fundamental parallelepiped* of L is
  $P(B) = \{a_1 v_1 + a_2 v_2 + \cdots + a_n v_n : a_i \in [0,1)\}$.
- **Notes**:
  1. Equivalently, $P(B) = \{Bx : x \in [0,1)^n\}$.
  2. $P(B)$ can be used to partition $\mathbb{R}^n$ into non-overlapping regions (called parallelepipeds). The "corners" of these parallelepipeds are the elements of the lattice $L(B)$.

Lattice: Example 2

- Let $n = 2$ and $B_2 = \{(2,0), (0,1)\}$.
- Then
  $L_2 = L(B_2) = \{B_2 x : x \in \mathbb{Z}^2\}$,
  where $B_2 = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$.
- **Notes**:

  1. L_2 a *sublattice* of L_1.

  2. $L_2 \neq L_1$ since $(1,0) \in L_1$, but
  $(1,0) = \frac{1}{2} \cdot (2,0) + 0 \cdot (0,1) \notin L_2$.



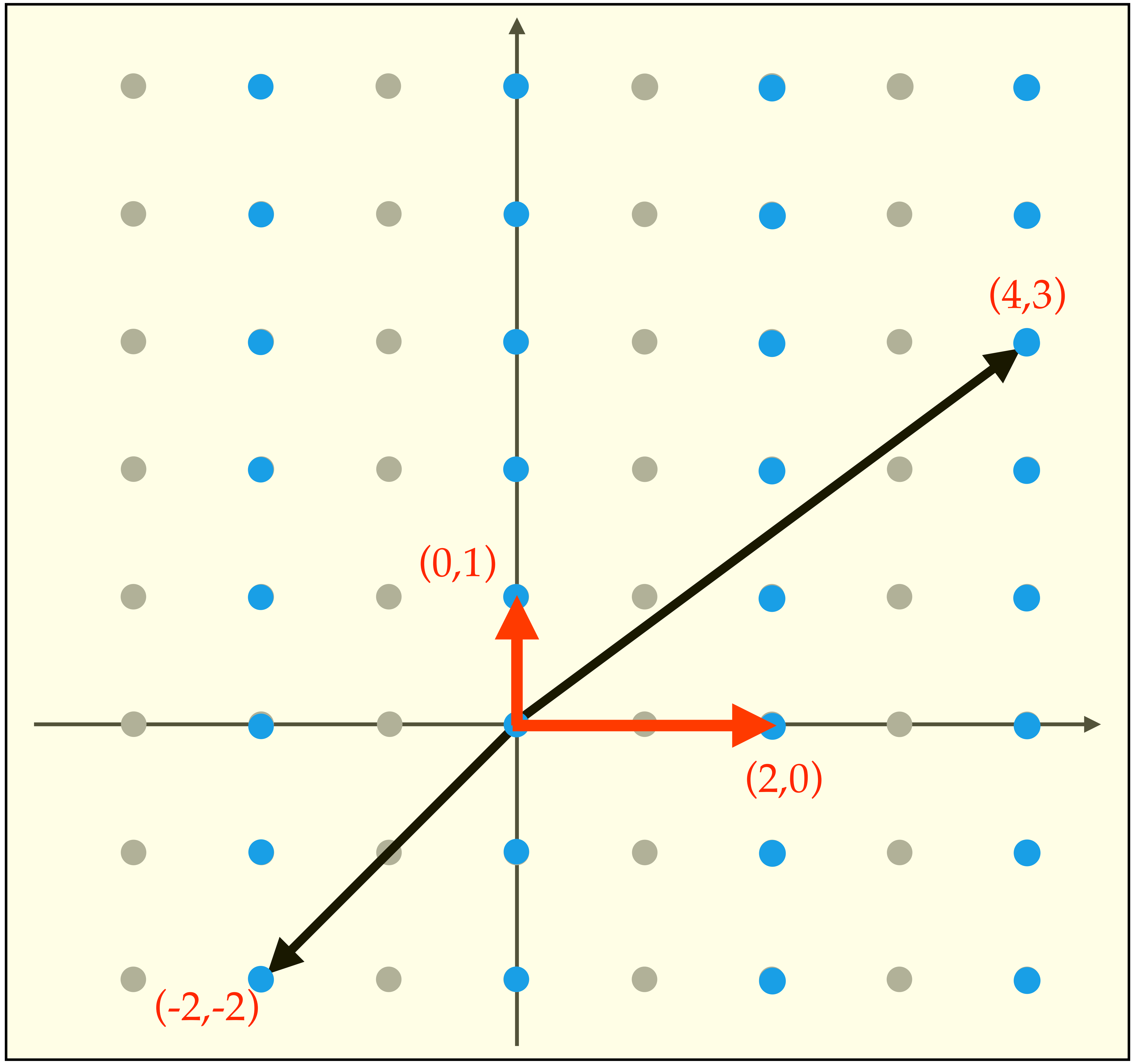
Lattice: Example 3

- Let $n = 2$ and $B_3 = \{(-2, -2), (4,3)\}$.
- Then $L_3 = L(B_3) = \{B_3 x : x \in \mathbb{Z}^2\}$, where $B_3 = \begin{bmatrix} -2 & 4 \\ -2 & 3 \end{bmatrix}$.
- **Notes**:

  1. $L_2 \subseteq L_3$ since
  $(2,0) = 3 \cdot (-2, -2) + 2 \cdot (4,3)$ and
  $(0,1) = -2 \cdot (-2, -2) - 1 \cdot (4,3)$.

  2. $L_3 \subseteq L_2$ since
  $(-2, -2) = -1 \cdot (2,0) - 2 \cdot (0,1)$ and
  $(4,3) = 2 \cdot (2,0) + 3 \cdot (0,1)$.

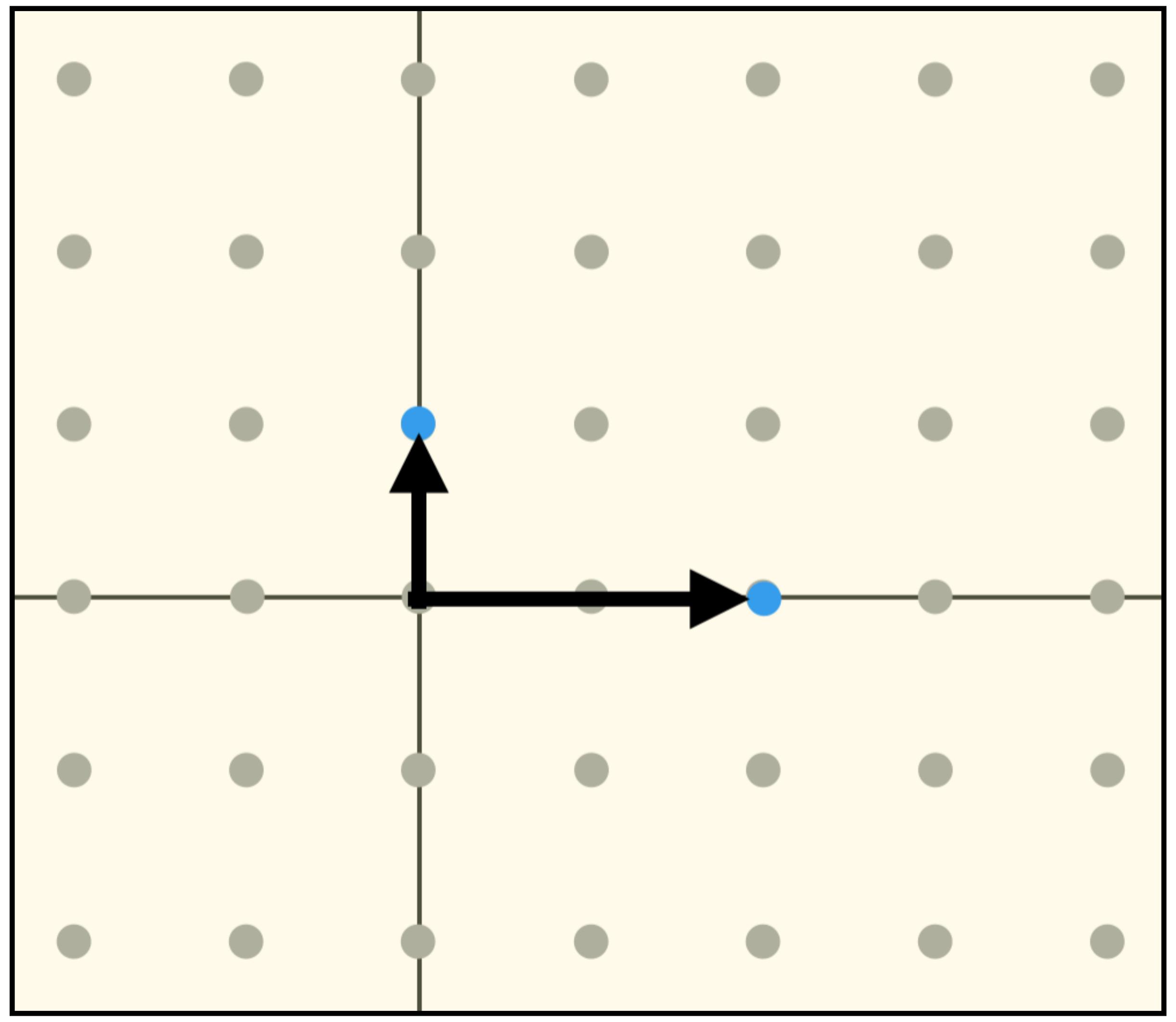
  3. Thus $L_3 = L_2$.



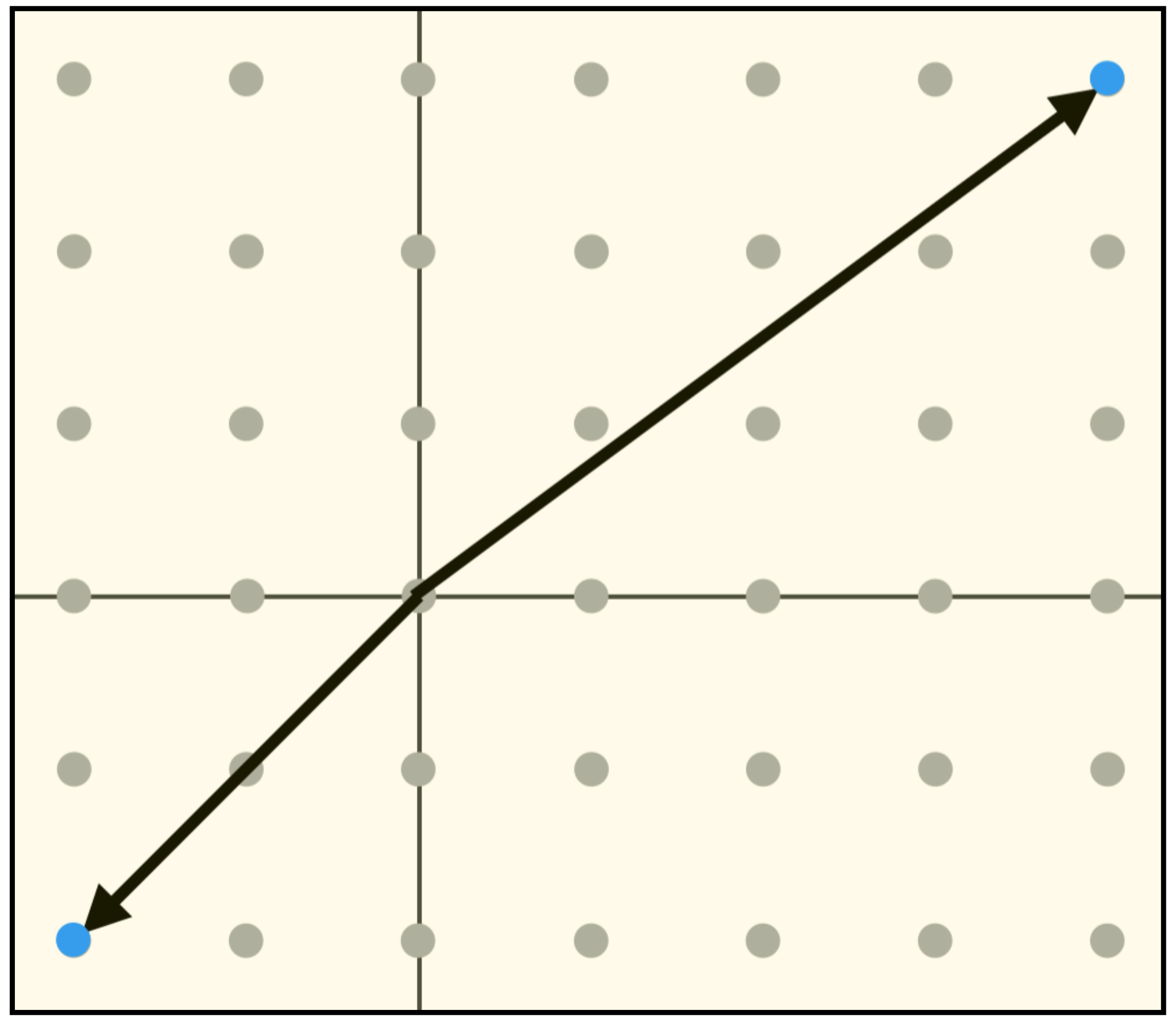
One basis is "nicer" than the other

- $L_2 = L(\{(2,0),(0,1)\}$ and $L_3 = L(\{(-2,-2),(4,3)\})$ are the same lattice, but described using different bases.
- The basis $B_2 = \{(2,0),(0,1)\}$ is "nicer" than the basis $B_3 = \{(-2,-2),(4,3)\}$ since the vectors in B_2 are "shorter" and "orthogonal" to each other.
- The *length* of a vector $a = (a_1, a_2, \ldots, a_n) \in \mathbb{R}^n$ is $\|a\|_2 = \sqrt{a_1^2 + a_2^2 + \cdots + a_n^2}$ (also called the *Euclidean length* or ℓ_2- *norm*).

B_2

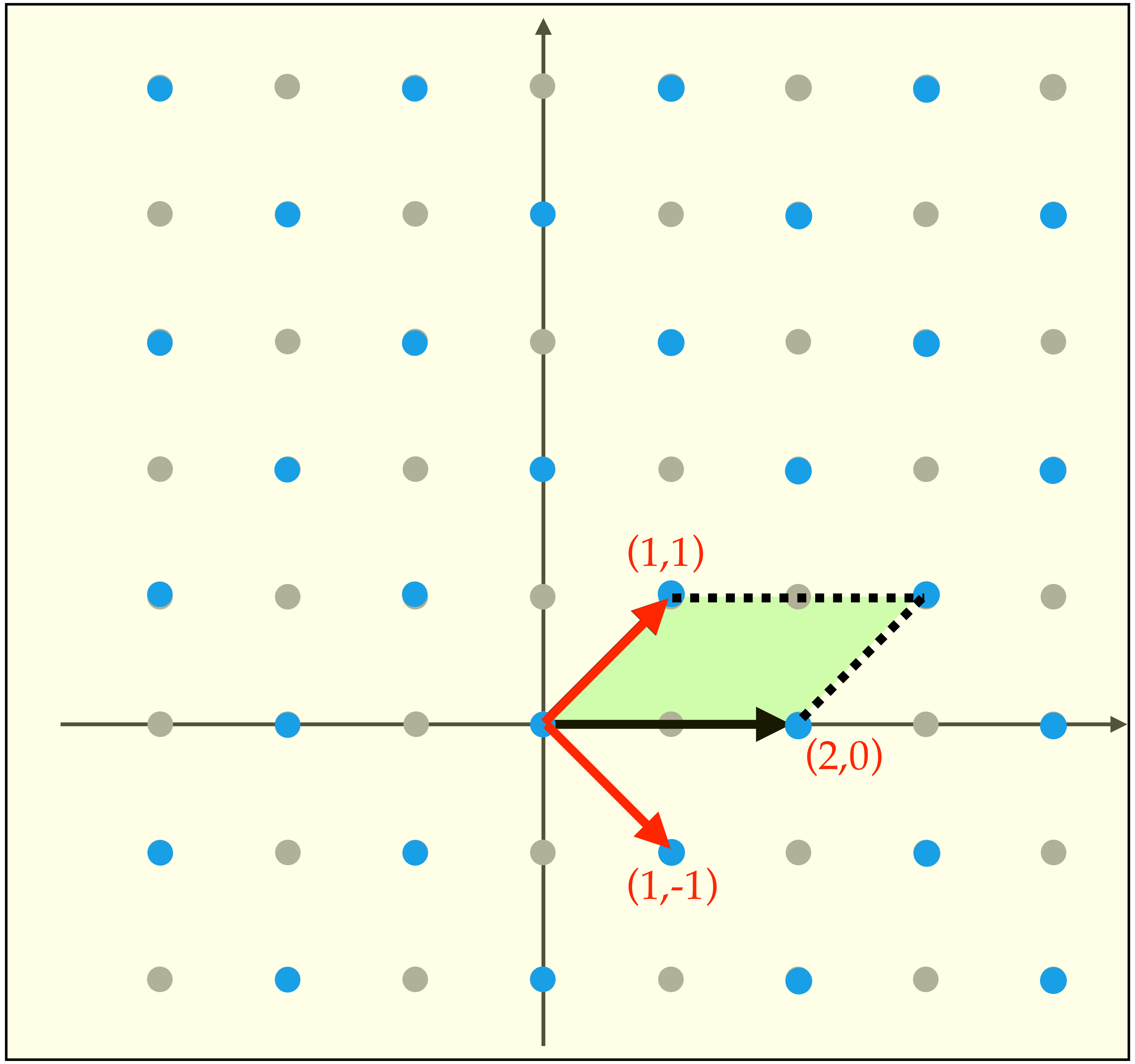


B_3



Lattice: Example 4

- Let $n = 2$ and $B_4 = \{(2,0), (1,1)\}$.
- Then $L_4 = L(B_4) = \{B_4 x : x \in \mathbb{Z}^2\}$, where $B_4 = \begin{bmatrix} 2 & 1 \\ 0 & 1 \end{bmatrix}$.
- Exercise: Prove that $L_4 \neq L_1$ and $L_4 \neq L_2$.
- Exercise: Prove that $\{(1,-1), (1,1)\}$ is another (nicer) basis for L_4.



A lattice has infinitely many bases

- **Theorem** (*characterization of lattice bases*) Let $L = L(B_1)$ be an n-dimensional (integer) lattice. Then an $n \times n$ integer matrix B_2 is also a basis for L if and only if $B_1 = B_2 U$, where U is an $n \times n$ matrix (the change-of-basis matrix) with integer entries and with $\det(U) = \pm 1$. (Such a matrix U is called *unimodular*.)

- **Example**. $B_2 = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$ and $B_3 = \begin{bmatrix} -2 & 4 \\ -2 & 3 \end{bmatrix}$ are bases for the same lattice since

$$\underbrace{\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}}_{B_2} = \underbrace{\begin{bmatrix} -2 & 4 \\ -2 & 3 \end{bmatrix}}_{B_3} \cdot \underbrace{\begin{bmatrix} 3 & -2 \\ 2 & -1 \end{bmatrix}}_{U}$$

where U is a unimodular matrix.

Proof of the characterization of lattice bases

- **Proof**. ($\Longrightarrow$) Suppose that B_1 and B_2 are both bases for $L \subseteq \mathbb{R}^n$.
  Since B_1 is a basis for L, and since the vectors in B_2 are in L, we can write $B_2 = B_1 U$ for some invertible matrix $U \in \mathbb{Z}^{n\times n}$.
  Similarly, we can write $B_1 = B_2 V$ for some invertible matrix $V \in \mathbb{Z}^{n\times n}$.
  Now, $B_1 = B_2 V = (B_1 U)V = B_1(UV)$.
  Since B_1 is invertible, we have $UV = I_n$.
  Thus, $\det(U)\det(V) = 1$, and hence $\det(U) = \pm 1$ and $\det(V) = \pm 1$.

  ($\Longleftarrow$) Exercise. $\square$

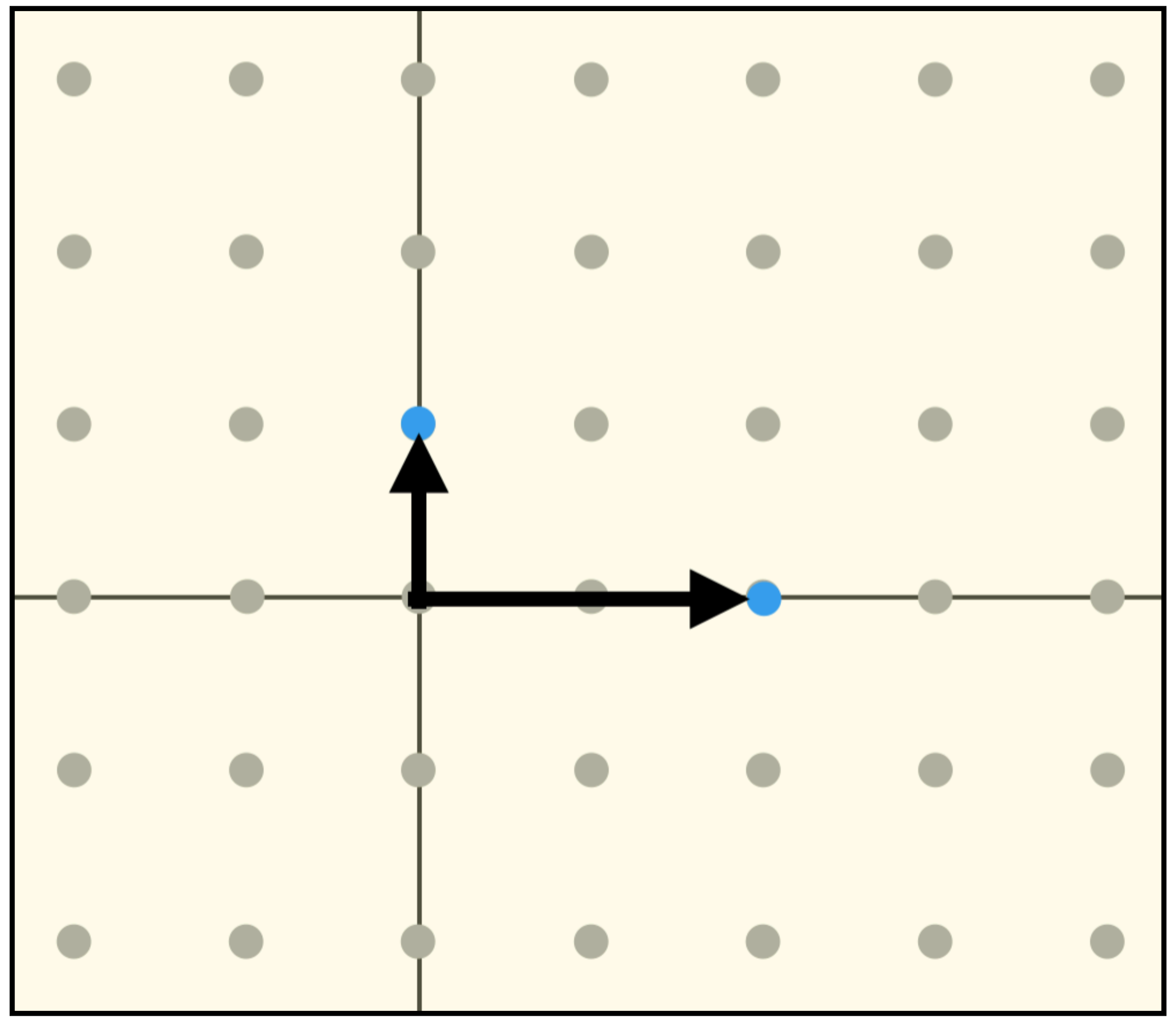
Volume of a lattice

- **Definition**. Let $L = L(B)$ be a lattice. The *volume* of L is $\text{vol}(L) = |\det(B)|$.
- **Note:** The volume of a lattice is the "volume" of the fundamental parallelepiped of the lattice.
  - If the lattice is 2-dimensional, then its volume is the *area* of its parallelepiped.
  - Informally, the volume of a lattice is inversely proportional to the density of its lattice vectors. The larger the volume, the sparser is the lattice.
- **Exercise.** Show that the volume is an *invariant* of L, i.e., it doesn't depend on the basis B chosen for L.
- **Exercise**. Suppose that $L_1 \subseteq L_2$. Prove that $\text{vol}(L_1) \geq \text{vol}(L_2)$.

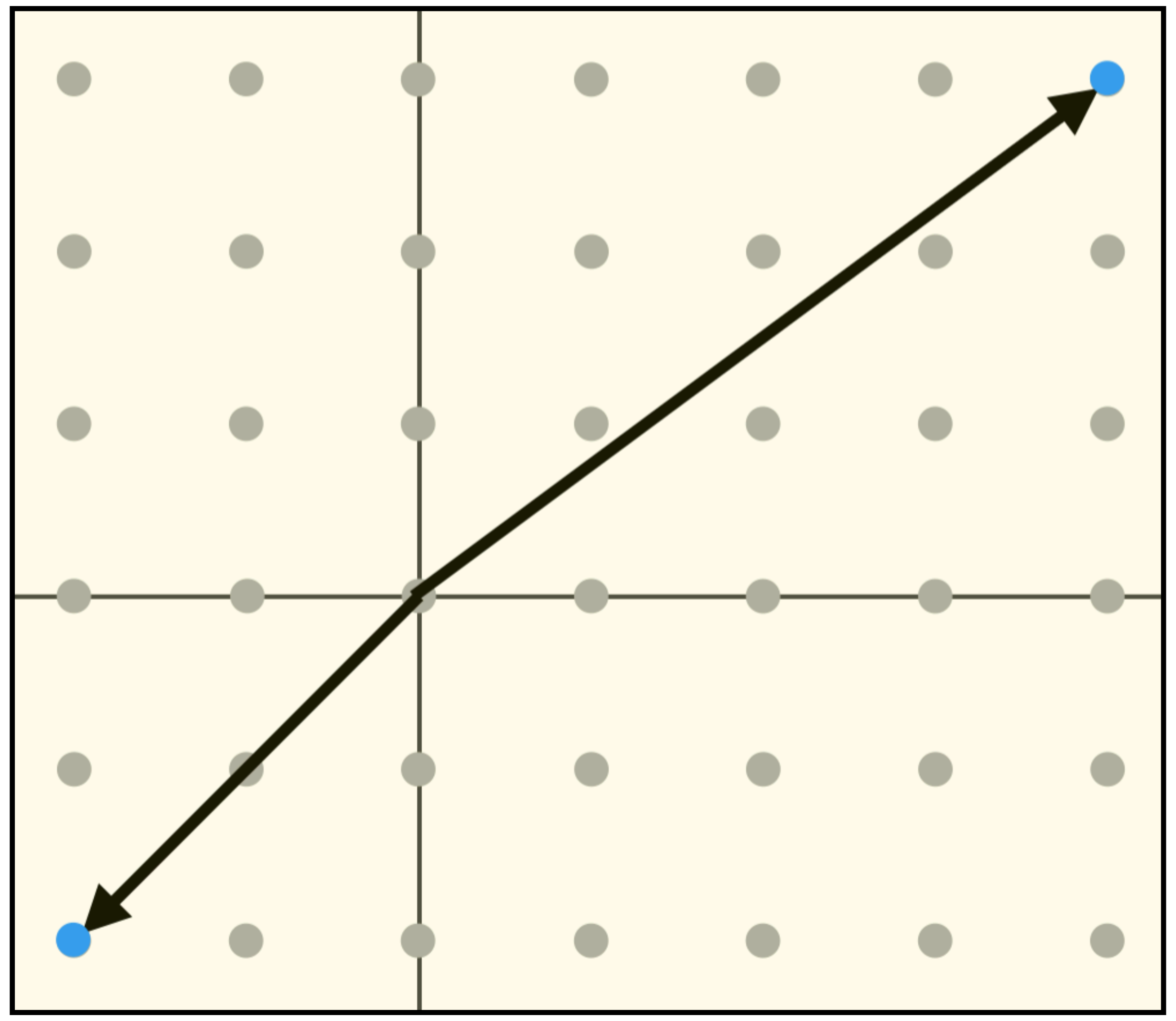
Some bases are nicer than others

- **Shortest Vector Problem (SVP)**: Given a lattice $L = L(B) \subseteq \mathbb{Z}^n$, find a shortest nonzero vector in L.
- **Example**: Consider the two SVP instances $L_2 = L(\{(2,0), (0,1)\})$ and $L_3 = L(\{(-2,-2), (4,3)\})$.
- So, hardness of an SVP instance $L(B)$ depends on the quality of the given basis B for L.

B_2



B_3



Successive minima

- A fundamental problem in lattice-based cryptanalysis is finding a "good" basis for a lattice.
- **Definition**: Let $L \subseteq \mathbb{Z}^n$ be a lattice. For each $i \in [1, n]$, the ith *successive minimum* $\lambda_i(L)$ is the smallest real number r such that L has i linearly independent vectors the longest of which has length r.
- **Notes**:

1. $\lambda_1(L) \leq \lambda_2(L) \leq \cdots \leq \lambda_n(L)$.
2. $\lambda_1(L)$ is the length of a shortest nonzero vector in L.
3. (*Minkowski's Theorem*) $\lambda_1(L) \leq \sqrt{n}\,\mathrm{vol}(L)^{1/n}$.
4. (*Gaussian Heuristic*) $\lambda_1(L) \approx \sqrt{n/(2\pi e)}\,\mathrm{vol}(L)^{1/n}$ for random lattices.
5. $\lambda_n(L)$ is a lower bound on the length of a shortest basis for L.

LLL lattice basis reduction algorithm

- (1982) The **Lenstra-Lenstra-Lovász (LLL) algorithm** is a polynomial-time algorithm for finding a relatively short basis for a lattice L.

- **Notes**:

1. The LLL algorithm is a clever modification of the Gram-Schmidt process for finding an orthogonal basis for a vector space in $\mathbb{R}^n$.

2. Let $B = \{b_1, b_2, \ldots, b_n\}$ be the basis for L produced by the LLL algorithm, with $\|b_1\|_2 \le \|b_2\|_2 \le \cdots \le \|b_n\|_2$. Then $\|b_i\|_2 \le 2^{(n-1)/2}\lambda_i(L)$ for $1 \le i \le n$.
In particular, $\|b_1\|_2 \le 2^{(n-1)/2}\lambda_1(L)$ and $\|b_n\|_2 \le 2^{(n-1)/2}\lambda_n(L)$.

3. Also, $\|b_1\|_2 \le 2^{(n-1)/4}\mathrm{vol}(L)^{1/n}$, and $\displaystyle\prod_{i=1}^{n} \|b_i\|_2 \le 2^{n(n-1)/4}\mathrm{vol}(L)$.

Cryptanalytic applications of LLL

- Let $B = \{b_1, b_2, \ldots, b_n\}$ be the basis for L produced by the LLL algorithm, with $\|b_1\|_2 \le \|b_2\|_2 \le \cdots \le \|b_n\|_2$. Then $\|b_i\|_2 \le 2^{(n-1)/2}\lambda_i(L)$ for $1 \le i \le n$.
- In practice, the basis produced by LLL is typically significantly shorter than the above guarantee.
- LLL has been used to design attacks on many number-theoretic problems and public-key cryptographic systems.
  - e.g., see “Lattice attacks on digital signatures schemes”, *Designs, Codes and Cryptography*, by N. Howgrave-Graham and N. Smart (2000): Finds the DSA (and ECDSA) secret key when a small number of bits of each per-message secret for several signed messages are leaked.
  - e.g., see “Lattice reduction in cryptology: an update”, *Proceedings of ANTS-IV*, by P. Nguyen and J. Stern (2000).

SVP: A fundamental lattice problem

- **Shortest Vector Problem (SVP)**: Given a lattice $L = L(B)$, find a lattice vector of length $\lambda_1(L)$.
  - SVP is **NP-hard**.
  - The fastest (classical) algorithm known for SVP has (heuristic) running time $2^{0.292n+o(n)}$.
  - The fastest quantum algorithm known for SVP has (heuristic) running time $2^{0.265n+o(n)}$.
- **Approximate-SVP problem (SVP$_\gamma$)**: Given a lattice $L = L(B)$, find a nonzero lattice vector of length at most $\gamma \cdot \lambda_1(L)$.
  - SVP$_\gamma$ is believed to be hard for small γ.
    It's NP-hard for constant γ, but likely isn't NP-hard if $\gamma > \sqrt{n}$.
  - For $\gamma = 2^k$, the fastest algorithm known for SVP$_\gamma$ has running time $2^{\tilde{\Theta}(n/k)}$
    (where $\tilde{\Theta}$ hides a power of $\log n$).
  - If $\gamma > 2^{(n \log\log n)/\log n}$, then SVP$_\gamma$ can be efficiently solved using the LLL algorithm.

SIVP: Another fundamental lattice problem

- **Shortest Independent Vectors Problem (SIVP)**: Given a lattice $L = L(B)$, find n linearly independent vectors in L all of which have length at most $\lambda_n(L)$.
  - A solution to SIVP isn't necessarily a basis for L.
  - SIVP is **NP-hard**.

- **Approximate-SIVP problem ($\mathbf{SIVP}_\gamma$)**: Given a lattice $L = L(B)$, find n linearly independent vectors in L all of which have length at most $\gamma \cdot \lambda_n(L)$.
  - The hardness of SIVP_γ is similar to that of SVP_γ.
  - **Fact**: $\text{SIVP}_{\gamma\sqrt{n}} \le \text{SVP}_\gamma$.