Finding shortest and closest vectors in a lattice of Voronoi's first kind

Robby McKilliam
Alex Grant
Vaughan Clarkson

Institute for Telecommunications Research
University of South Australia

School of Information Technology and Electrical Engineering
The University of Queensland

May 26, 2014

Lattices

An n-dimensional **lattice** Λ is a discrete set of vectors from $\mathbb{R}^m$, $m \geq n$, given by

$$\Lambda = \{b_1 u_1 + b_2 u_2 + \cdots + b_n u_n \mid u_1, \ldots u_n \in \mathbb{Z}\},$$

where $b_1, \ldots, b_n \in \mathbb{R}^m$ are **basis vectors** of Λ.

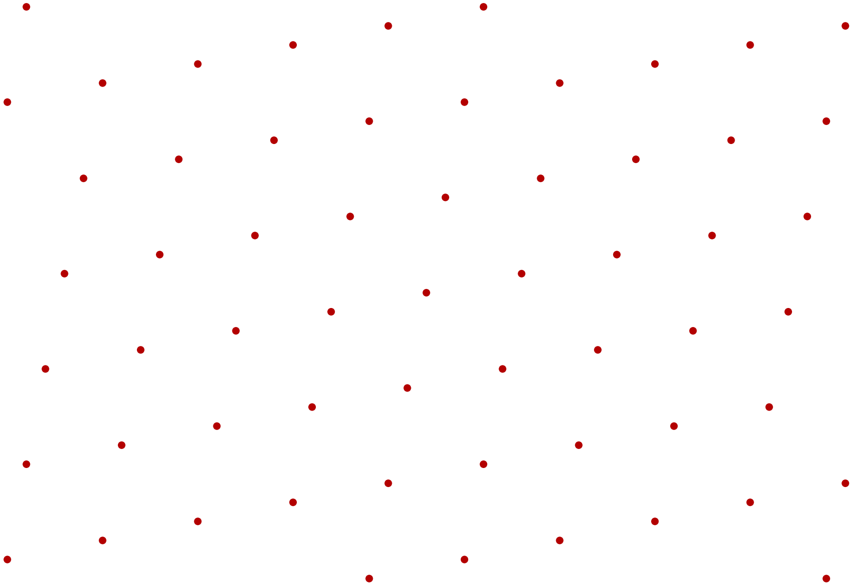

Figure : A 2-dimensional lattice.

Short vectors

Those lattice points with smallest non-zero length are called **short vectors**. That is, the short vectors have squared length

$$\min_{x \in \Lambda \setminus \{0\}} \|x\|^2.$$

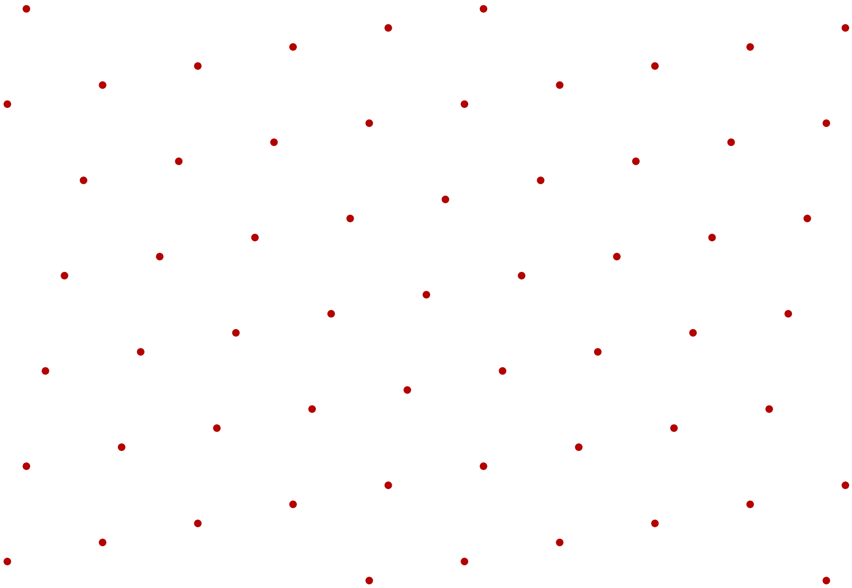

Figure : A 2-dimensional lattice.

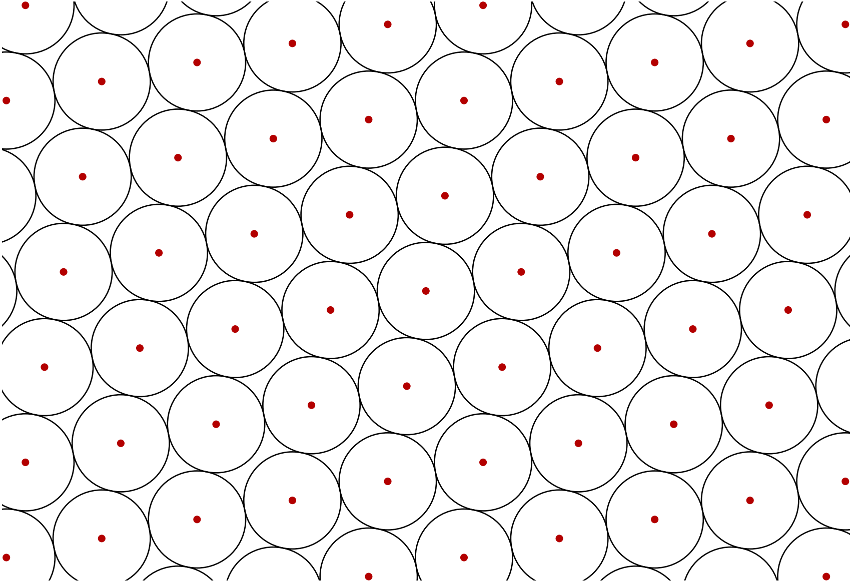

Figure : A 2-dimensional lattice. There are 4 short vectors.

The shortest vector problem

- Computing a short vector is called the **shortest vector problem**.
- Applications in cryptography and number theory.
- NP-hard for arbitrary lattices.
- Easier for specific lattices.
- Short vectors are easy to find in the **root lattices** $\mathbb{Z}^n$, A_n, and D_n.
- We will show that the problem is relatively easy to solve for lattices of **Voronoi's first kind**.

The Voronoi cell

The **Voronoi cell** of a lattice $\Lambda \subset \mathbb{R}^m$ is the subset of $\mathbb{R}^m$ at least as close to the origin than to any lattice point,

$$\mathrm{Vor}(\Lambda) = \{x \in \mathbb{R}^m \mid \|x\| \leq \|x - y\|, y \in \Lambda\}.$$

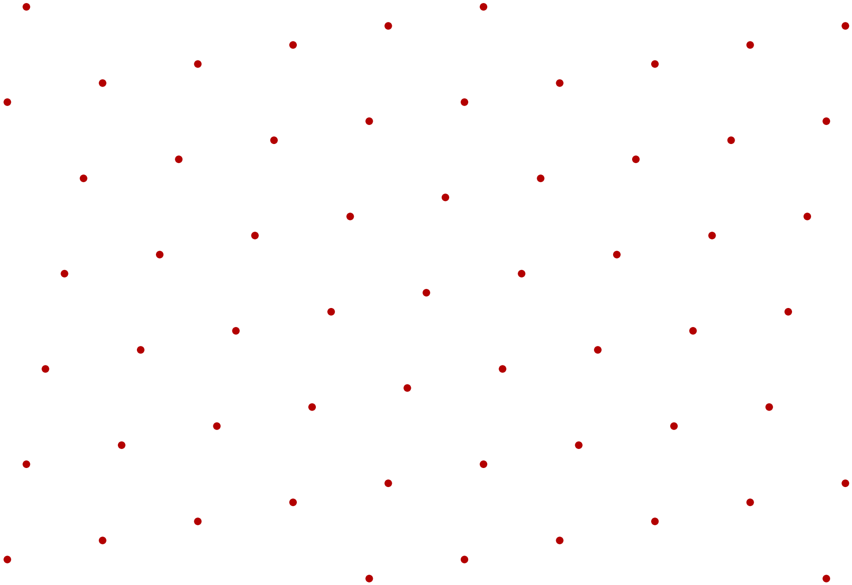

Figure : A 2-dimensional lattice and its Voronoi cell.

Relevant vectors

The **relevant vectors** of a lattice Λ are those which contribute a face to the Voronoi cell.

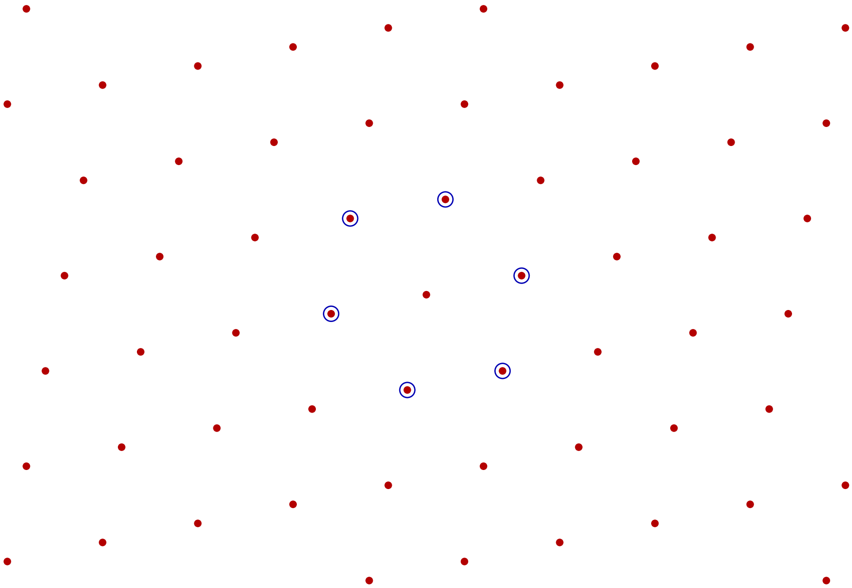

Figure : A 2-dimensional lattice with 6 relevant vectors.

Relevant vectors

- Denote the set of relevant vectors by $\mathrm{Rel}(\Lambda)$.
- The Voronoi cell can be defined using the relevant vectors,

$$\mathrm{Vor}(\Lambda) = \{x \in \mathbb{R}^m \mid \|x\| \leq \|x - v\|, v \in \mathrm{Rel}(\Lambda)\}.$$

- Short vectors are relevant vectors.

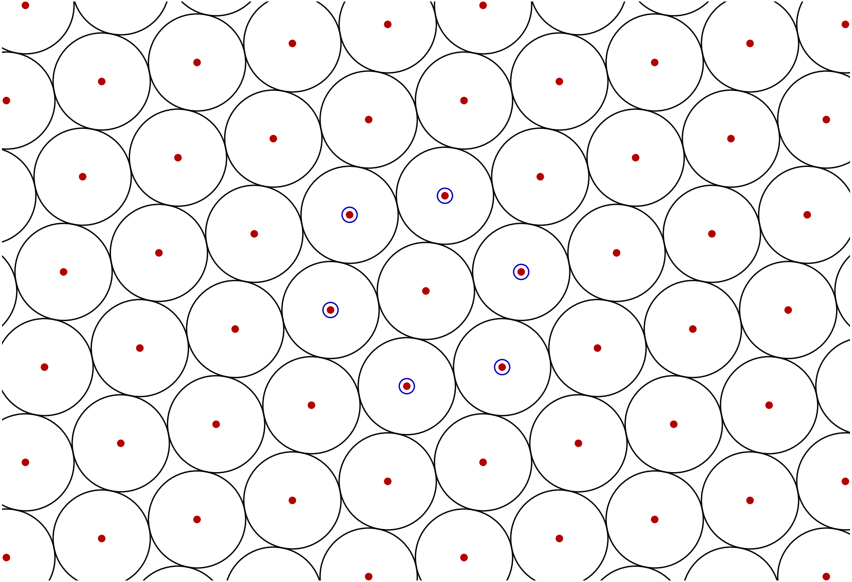

Figure : A lattice with 6 relevant vectors and 4 short vectors.

The closest lattice point problem

Given a lattice $\Lambda \subset \mathbb{R}^m$ and a vector $y \in \mathbb{R}^m$ find $x \in \Lambda$ such that

$$\|y - x\|^2$$

is minimised.

- This is called the **closest lattice point problem** and a solution is called a **closest lattice point** to y.
- The lattice point $x \in \Lambda$ is closest to $y \in \mathbb{R}^m$ if and only if

$$y \in \mathrm{Vor}(\Lambda) + x.$$

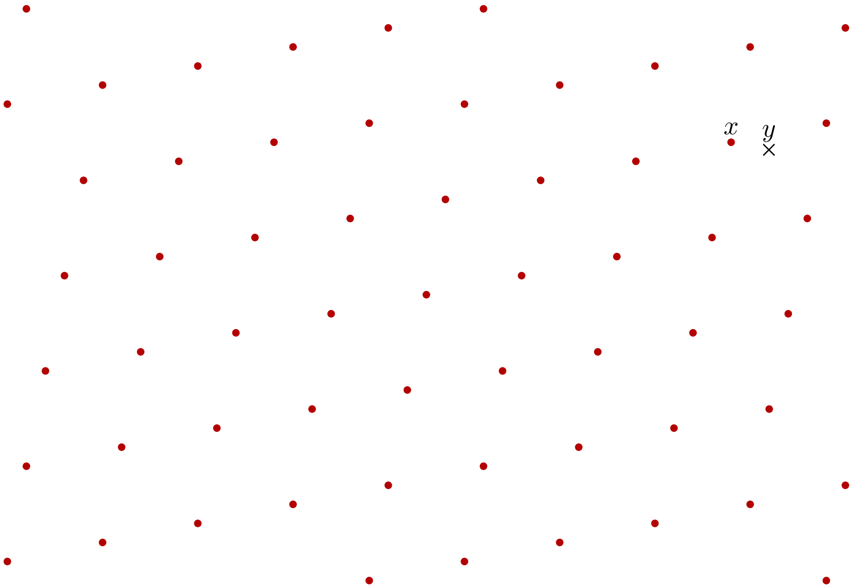

Figure : The closest lattice point.

The closest lattice point problem

Applications to:

- coding and quantisation,
- multi-antenna communication (MIMO),
- unwrapping of phase data for electronic distance measurement in GPS and surveying,
- single frequency estimation,
- polynomial phase estimation,
- circular statistics.

The closest lattice point problem

- NP-hard for arbitrary lattices.
- Easier for specific lattices.
- Fast algorithms exist for the **root lattices** $\mathbb{Z}^n$, A_n, and D_n.
- We will describe a fast algorithm to compute a closest point in lattices of **Voronoi's first kind**.

Lattices of Voronoi's first kind

An n-dimensional lattice Λ is of **Voronoi's first kind** if it has an **obtuse superbase**, that is, a set of $n+1$ vectors

$$b_1, \dots, b_{n+1}$$

such that

- $b_1, \dots, b_n$ are a basis for Λ,
- $b_1 + b_2 \cdots + b_{n+1} = 0 \quad$ (the superbase condition),
- $q_{ij} = b_i \cdot b_j \leq 0$ whenever $i \neq j \quad$ (the obtuse condition).

The q_{ij} are called **Selling parameters**.

An example

Consider the 3-dimensional lattice with basis

$$\begin{aligned} b_1 &= \left[\begin{array}{ccc} 2 & -1 & 0 \end{array} \right] \\ b_2 &= \left[\begin{array}{ccc} -1 & 2 & 0 \end{array} \right] \\ b_3 &= \left[\begin{array}{ccc} 0 & 0 & 2 \end{array} \right]. \end{aligned}$$

Define a 4th vector as

$$b_4 = -b_1 - b_2 - b_3 = \left[\begin{array}{ccc} -1 & -1 & -2 \end{array} \right],$$

so that b_1, b_2, b_3, b_4 satisfy the superbase condition.

An example

The Selling parameters can be written in a matrix

$$\begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ q_{31} & q_{32} & q_{33} & q_{34} \\ q_{41} & q_{42} & q_{43} & q_{44} \end{bmatrix} = \begin{bmatrix} 5 & -4 & 0 & -1 \\ -4 & 5 & 0 & -1 \\ 0 & 0 & 4 & -4 \\ -1 & -1 & -4 & 6 \end{bmatrix}.$$

The off diagonal elements are not positive so the obtuse condition is satisfied.

Lattices of Voronoi's first kind

Theorem (Conway and Sloane (1992))

Let Λ be a n-dimensional lattice of Voronoi's first kind with obtuse superbase $b_1, \ldots, b_{n+1}$. The relevant vectors in Λ are of the form

$$\sum_{i \in I} b_i$$

where I is a strict subset of $\{1, 2, \ldots, n+1\}$ and I is not empty.

Corollary

Short vectors in Λ are of the form $\sum_{i \in I} b_i$.

Lattices of Voronoi's first kind

A naïve way to compute a short vector is to compute

$$\|\sum_{i \in I} b_i\|^2$$

for all of the $2^{n+1} - 2$ possible subsets I.

- Requires a number of operations that grows exponentially with the dimension n.
- We can improve this using a **minimum cut algorithm**.

Graphs, cuts, and minimum cuts

Let G be a weighted graph with:

- $n + 1$ vertices $v_1, \ldots, v_{n+1}$,
- edges e_{ij} connecting vertex v_i to vertex v_j,
- edge weights $w_{ij} \in \mathbb{R}$.

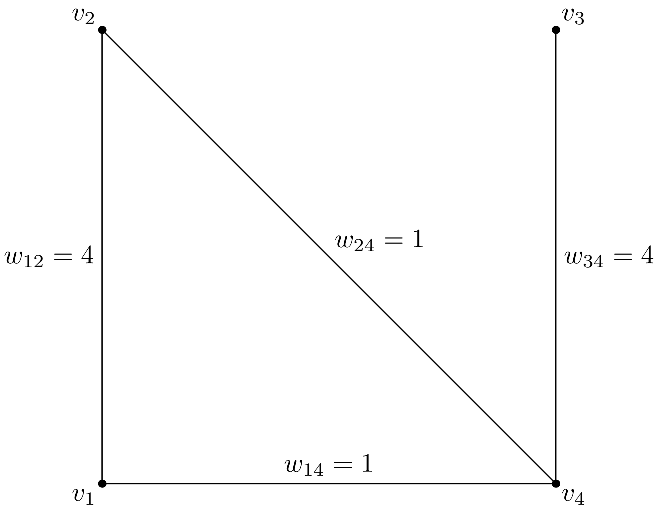

Figure : A graph with 4 vertices and 4 weighted edges.

Graphs, cuts, and minimum cuts

A **cut** in G is a partition of the vertices into two nonempty sets C and its complement $\bar{C}$.

- The **weight** of a cut is the sum of the weights on the edges crossing from the vertices in C to the vertices in $\bar{C}$.
- A **minimum cut** is a pair $(C, \bar{C})$ with smallest weight.

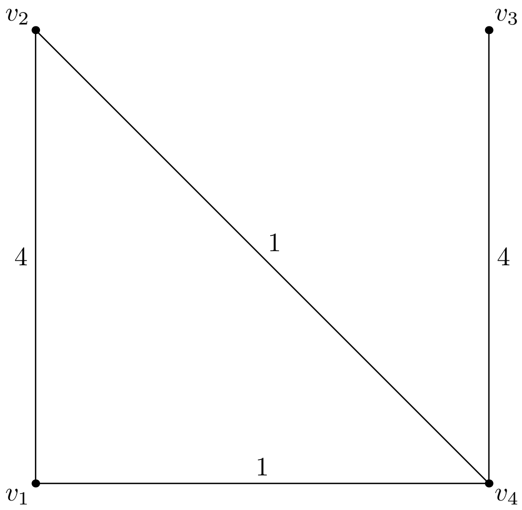

Figure : A graph with 4 vertices and 4 weighted edges.

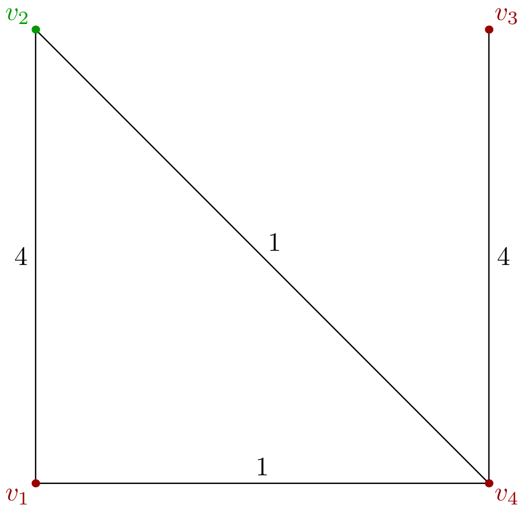

Figure : The cut $C = \{v_2\}$ and $\bar{C} = \{v_1, v_3, v_4\}$

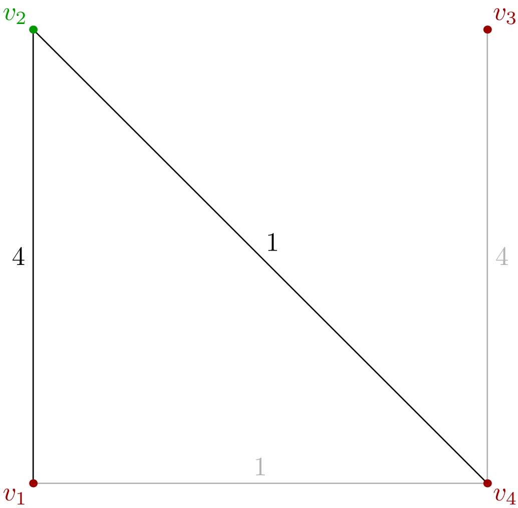

Figure : The cut $C = \{v_2\}$ and $\bar{C} = \{v_1, v_3, v_4\}$ has weight 5.

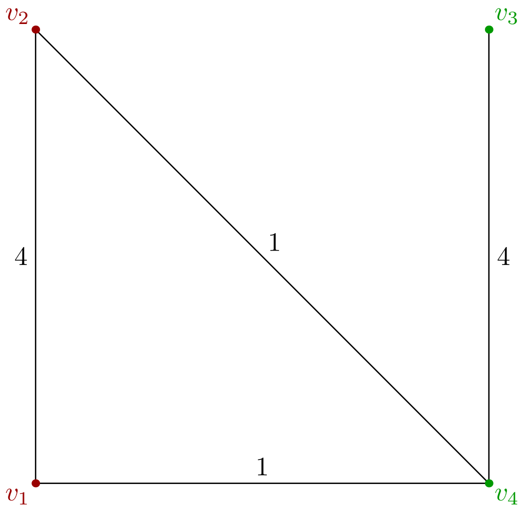

Figure : The minimum cut $C = \{v_3, v_4\}$ and $\bar{C} = \{v_1, v_2\}$

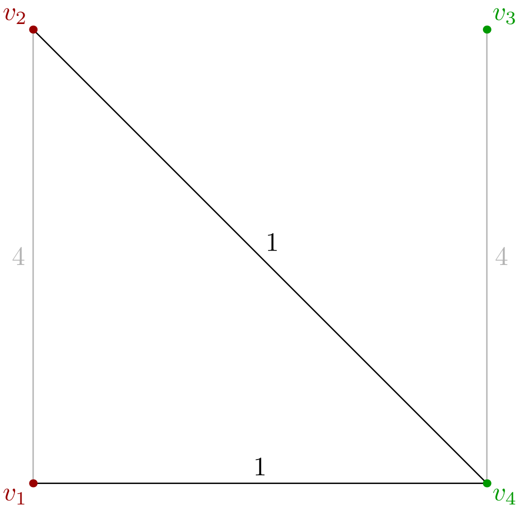

Figure : The minimum cut $C = \{v_3, v_4\}$ and $\bar{C} = \{v_1, v_2\}$ has weight 2.

Graphs, cuts, and minimum cuts

If the edge weights w_{ij} are all nonnegative, a minimum cut can be computed:

- deterministically in $O(n^3)$ operations using the algorithm of Stoer and Wagner (1997),
- with high probability in $O(n^2 \log(n)^3)$ operations using the randomised algorithm of Karger and Stien (1996).

Theorem (McKilliam and Grant (2012))

Let Λ be a n-dimensional lattice of Voronoi's first kind with obtuse superbase

$$b_1, \ldots, b_{n+1}.$$

Let G be a graph with $n+1$ vertices $v_1, \ldots, v_{n+1}$ and edge weights

$$w_{ij} = -q_{ij} = -b_i \cdot b_j \geq 0 \qquad i \neq j.$$

Let $(C, \bar{C})$ be a minimum cut in G. A short vector in Λ is

$$\sum_{i \in I} b_i \quad \textit{where} \quad I = \{i \mid v_i \in C\}.$$

The squared length of the short vector is given by the weight of the minimum cut.

An example

Consider again the 3-dimensional lattice with obtuse superbase

$$\begin{aligned} b_1 &= \begin{bmatrix} 2 & -1 & 0 \end{bmatrix} \\ b_2 &= \begin{bmatrix} -1 & 2 & 0 \end{bmatrix} \\ b_3 &= \begin{bmatrix} 0 & 0 & 2 \end{bmatrix} \\ b_4 &= \begin{bmatrix} -1 & -1 & -2 \end{bmatrix}. \end{aligned}$$

The Selling parameters are given in matrix form as

$$\begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ q_{31} & q_{32} & q_{33} & q_{34} \\ q_{41} & q_{42} & q_{43} & q_{44} \end{bmatrix} = \begin{bmatrix} 5 & -4 & 0 & -1 \\ -4 & 5 & 0 & -1 \\ 0 & 0 & 4 & -4 \\ -1 & -1 & -4 & 6 \end{bmatrix}.$$

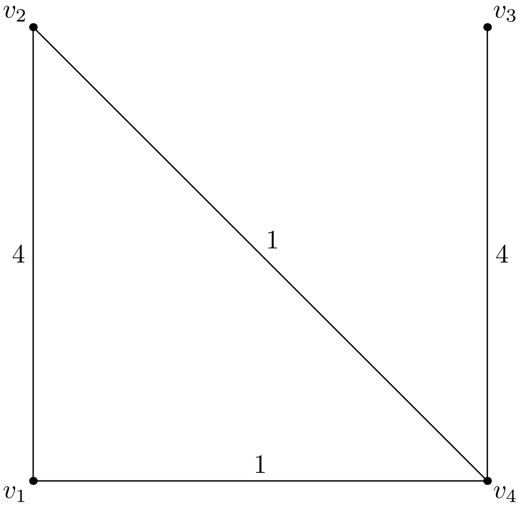

Figure : We have seen this graph before!

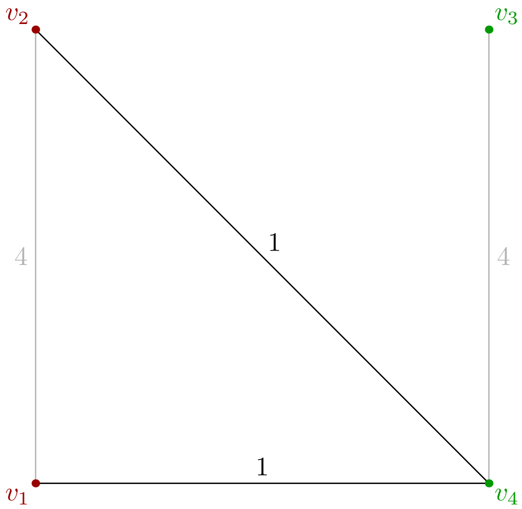

Figure : The minimum cut $C = \{v_3, v_4\}$ and $\bar{C} = \{v_1, v_2\}$ has weight 2.

An example

The minimum cut corresponds with the short vectors

$$b_1 + b_2 = [1, 1, 0]$$

and

$$b_3 + b_4 = -b_1 - b_2 = [-1, -1, 0]$$

of squared length 2.

Some questions we asked in 2012

- Can we efficiently decide whether a lattice is of Voronoi's first kind?
- Can we efficiently find an obtuse superbase if it exists?
- Can a similar approach be taken to solve the **closest lattice point problem**?

Some questions we asked in 2012

- Can we ~~efficiently~~ decide whether a lattice is of Voronoi's first kind? **Yes**
- Can we ~~efficiently~~ find an obtuse superbase if it exists? **Yes**
- Can a similar approach be taken to solve the **closest lattice point problem**?

Some questions we asked in 2012

- Can we ~~efficiently~~ decide whether a lattice is of Voronoi's first kind? **Yes**
- Can we ~~efficiently~~ find an obtuse superbase if it exists? **Yes**
- Can a similar approach be taken to solve the **closest lattice point problem**? $O(n^4)$

A series of relevant vectors

Let x_0 be some lattice point from Λ and consider the following iteration,

$$\begin{aligned} x_{k+1} &= x_k + v_k \\ v_k &= \arg \min_{v \in \mathrm{Rel}(\Lambda) \cup \{0\}} \|y - x_k - v\| \end{aligned}$$

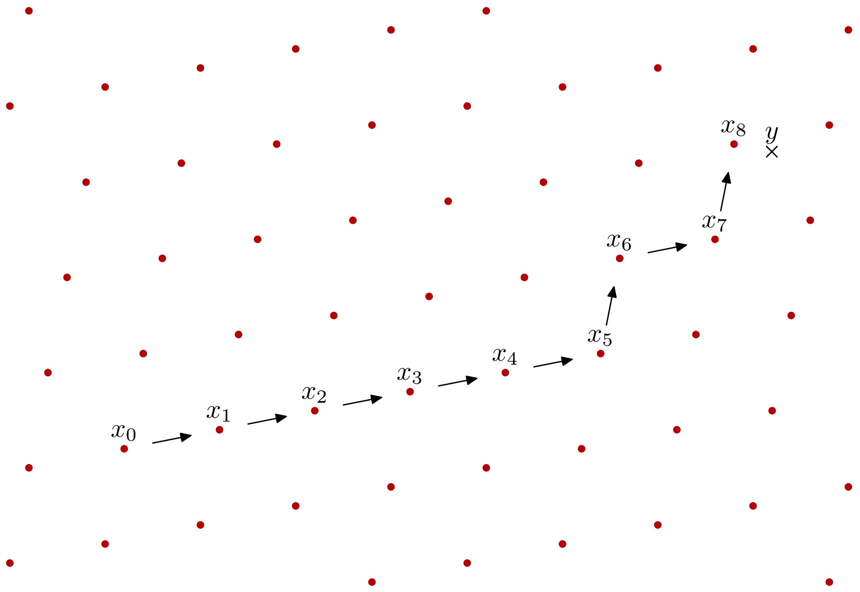

Figure : Computing a closest point by a series of relevant vectors.

A series of relevant vectors

- The number of iterations depends on x_0 and might be large.
- Minimising over the set of relevant vectors, that is computing

$$\arg \min_{v \in \mathrm{Rel}(\Lambda) \cup \{0\}} \|y - x_k - v\|$$

  might be expensive.
- There are as many as $2^{n+1} - 2$ relevant vectors.

A series of relevant vectors

For a lattice of Voronoi's first kind:

- x_0 can be chosen to ensure that a closest lattice point is found after at most n iterations.
- Minimisation over the set of relevant vectors can be performed by computing a minimum cut in a flow network.
- Using known algorithms a minimum cut can be found in $O(n^3)$ operations.
- In total $O(n^4)$ operations are required to compute a closest lattice point.

Theorem (McKilliam, Grant, Clarkson (2014))

Let Λ be a n-dimensional lattice of Voronoi's first kind with obtuse superbase $b_1, \ldots, b_{n+1}$. Let $z_1, \ldots, z_{n+1} \in \mathbb{R}$ minimise

$$\|y - \sum_{i=1}^{n+1} b_i z_i\|$$

and put

$$x_0 = \sum_{i=1}^{n+1} b_i \lfloor z_i \rfloor .$$

The iterative procedure, initialized at x_0, converges to a closest lattice point in at most n iterations.

What now?

- Can good codes or quantisers be constructed from lattices of Voronoi's first kind?
- Do applications such as global positioning, phase unwrapping, etc., involve lattices of Voronoi's first kind?
- Are there subfamilies of Voronoi's first kind that admit even faster algorithms?
- Are there other families of lattices for which similar techniques lead to polynomial time algorithms?