Low-Dimensional Lattice Basis Reduction Revisited

Damien STEHLÉ

Burlington, June 17th, 2004

Joint work with Phong NGUYEN

`http://www.loria.fr/~stehle/`

`damien.stehle@loria.fr`

Lattices

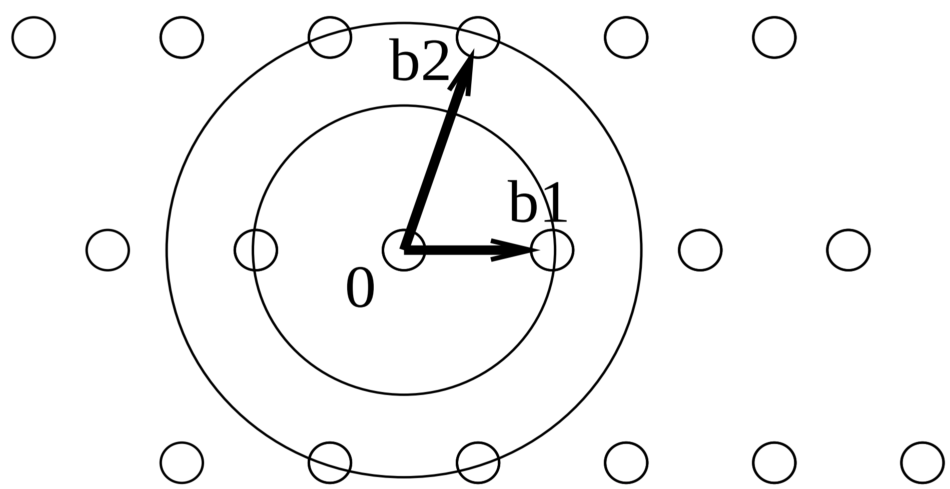
- Lattice L = grid in a Euclidean space
  $= $ discrete subgroup of $\mathbb{R}^d$
  $= \{\sum_{i=1}^{m} x_i \mathbf{b}_i \mid x_1, \ldots, x_m \in \mathbb{Z}\}$.
- d is the space dim, $m \leq d$ the dim, $[\mathbf{b}_1, \ldots, \mathbf{b}_m]$ a basis.
- L given by the integer matrix of one of its bases, along with its Gram matrix:

$$G(\mathbf{b}_1, \ldots, \mathbf{b}_m) = (\langle \mathbf{b}_i, \mathbf{b}_j \rangle)_{i,j}.$$

- Complexity model: bit operations, without fast arithmetic.

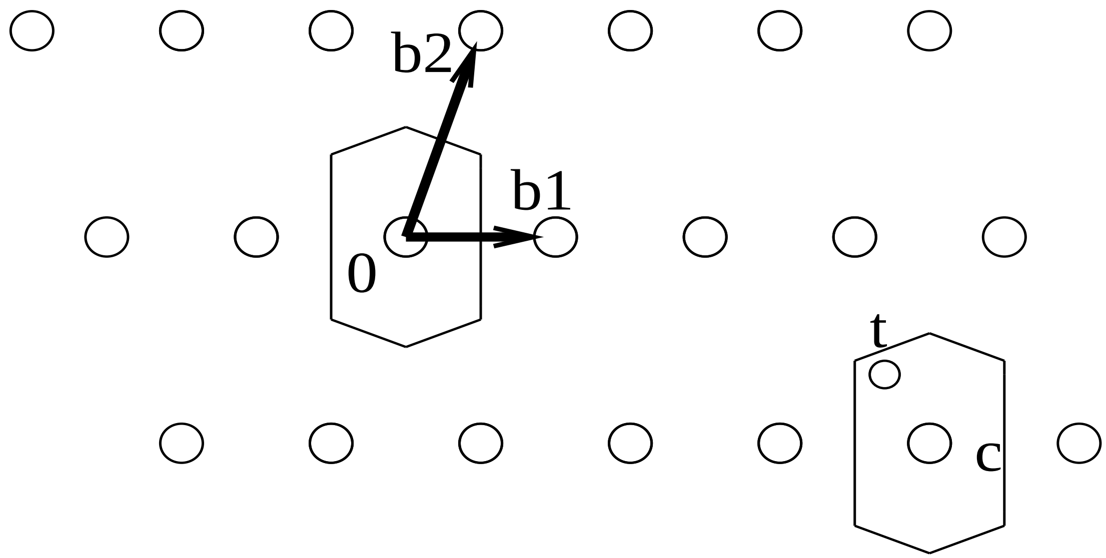
Basic Definitions (1/2)

- First minimum $= \lambda_1(L) = \min(r \mid B_n(\mathbf{0}, r) \cap L \neq \{\mathbf{0}\})$.
- SVP: find $\mathbf{v} \in L$ of length $\lambda_1(L)$.
- i-th minimum $= \lambda_i(L) = \min(r \mid B_n(\mathbf{0}, r) \cap L$ has dim $\geq i)$.

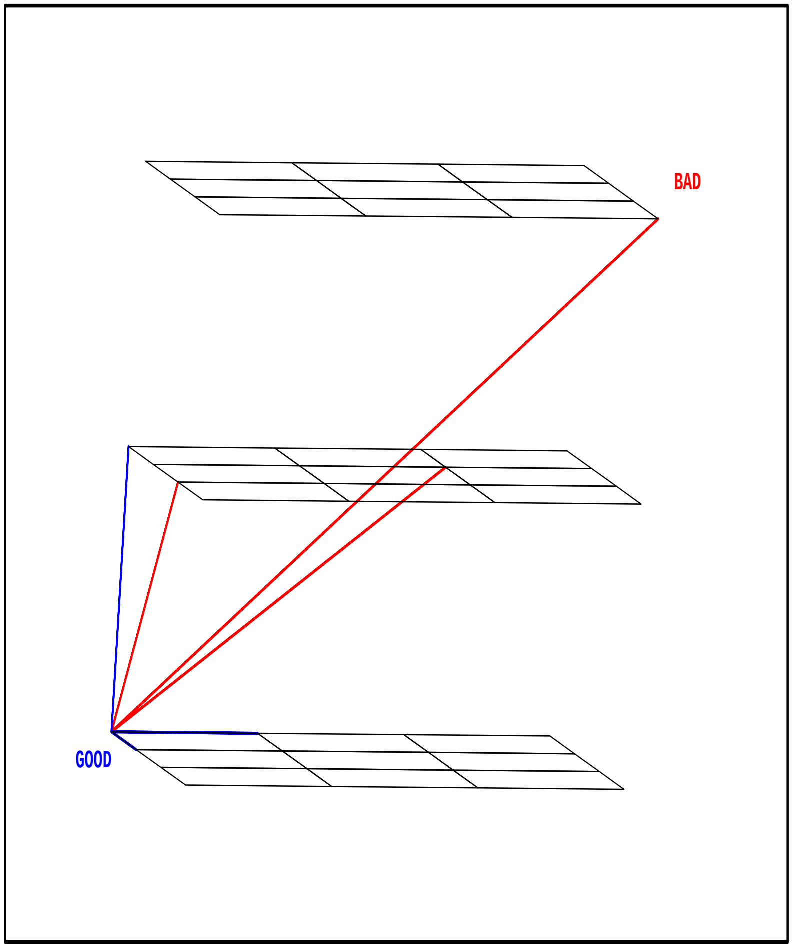


Basic Definitions (2/2)

- Voronoï cell $= \mathrm{Vor}(L)$
  $= \{\mathbf{x} \in \mathrm{Span}_{1..m}(\mathbf{b}_i) \mid \forall \mathbf{b} \in L, \|\mathbf{x}-\mathbf{b}\| \geq \|\mathbf{x}-\mathbf{0}\|\}$.
- CVP: Given $\mathbf{t} \in \mathbb{R}^d$, find $\mathbf{c} \in L$ s.t. $\mathbf{t} \in \mathbf{c} + \mathrm{Vor}(L)$.



Lattice Basis Reduction (1/2)



Lattice Basis Reduction (2/2)

- There are more or less interesting bases for a given lattice.
- Quality measures: lengths and orthogonality of the vectors.
- No natural "best" reduction.

- $[\mathbf{b}_1, \ldots, \mathbf{b}_m]$ is Minkowski (M-)reduced iff for any i, $\mathbf{b}_i$ is a shortest lattice vector s.t. $[\mathbf{b}_1, \ldots, \mathbf{b}_i]$ can be extended to a basis.
- If $d \leq 4$, a M-reduced basis reaches the d first minima.

Why Lattices in Low Dimensions?

- Gcd calculation in $\mathcal{O}_d$ (Kaltofen and Rolletschek).
- Sum of 4 squares.
- Rational points on rational conics (Cremona and Rusin).

- High dim lattice reduction relies on alg. in low dim (LLL, BKZ).
- Good starting point to a better understanding of lattices.
- Very elegant problem.

Some Bibliography

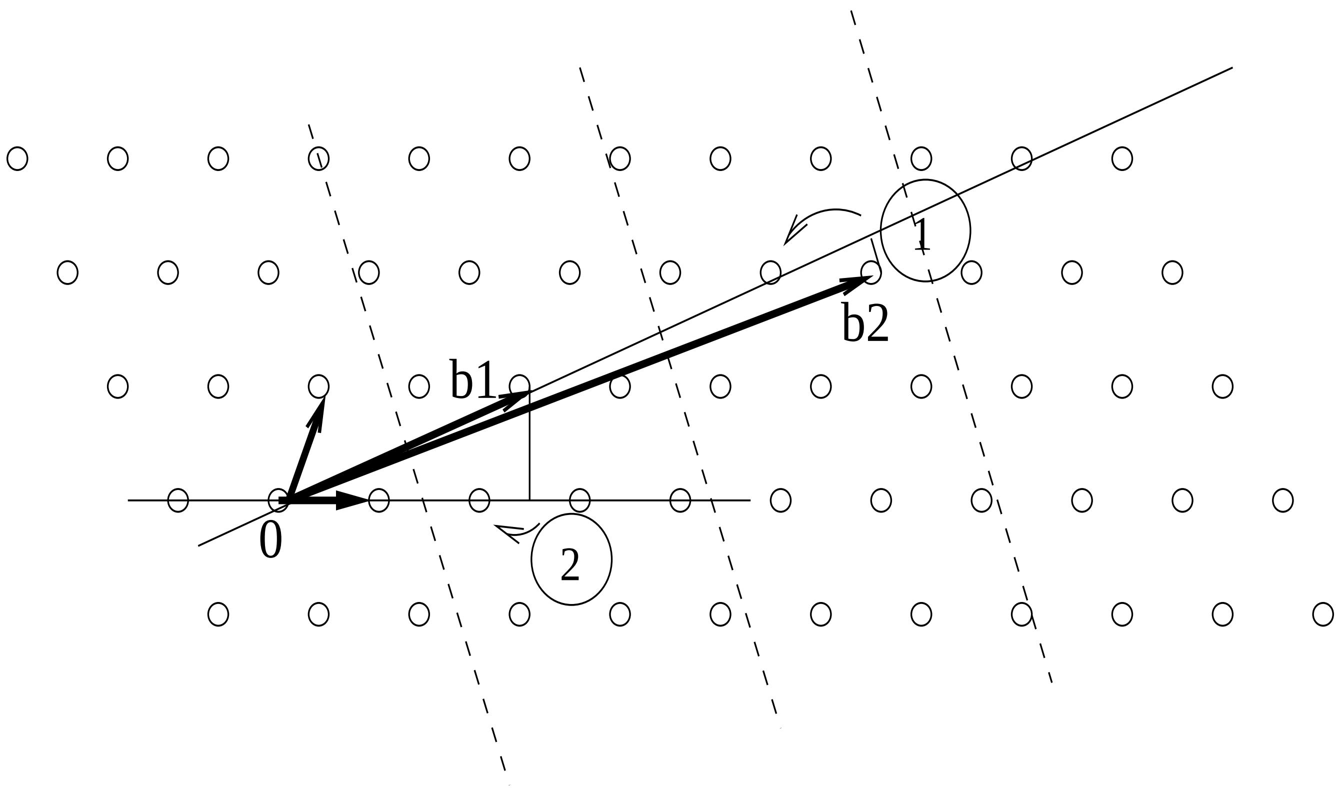
Fixed dimension, complexity with respect to the size of the matrix coefficients.

- 19-th c.: Gauss' algorithm in dim 2, quadratic complexity.
- 1982-83: LLL and Kannan, cubic complexity in any dim.
- 1986: "Affine" algorithm of Vallée in dim 3, cubic complexity.
- 1987: Schnorr's BKZ algorithm.
- 2001: Semaev's algorithm in dim 3, quadratic complexity.

Our Results

- Description of a natural greedy algorithm generalizing Gauss' and Semaev's algorithms.
- Proof that it returns a M-reduced basis in any dimension $d \leq 4$.
- Proof that it has a quadratic complexity in any dimension $d \leq 4$.
- Unified geometric analysis for all dimensions up to 4.

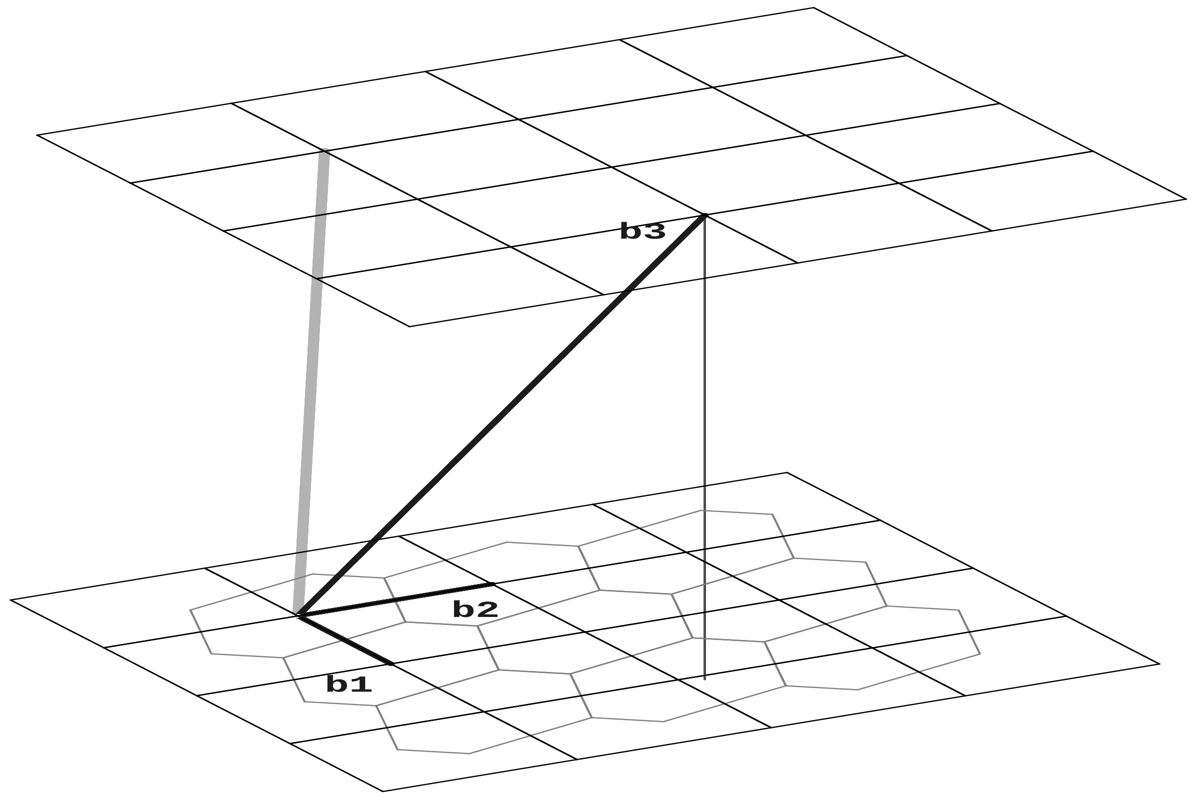
Gauss' Algorithm (1/2)



Gauss' Algorithm (2/2)

- Correctness: if $||\mathbf{b}_1|| \leq ||\mathbf{b}_2||$ and $\forall x \in \mathbb{Z}, ||\mathbf{b}_2 + x\mathbf{b}_1|| \geq ||\mathbf{b}_2||$, then $[\mathbf{b}_1, \mathbf{b}_2]$ is M-reduced.
- Linearity of the number of loop iterations: at least once in every 2 iterations, we subtract $x\mathbf{b}_1$ to $\mathbf{b}_2$ with $|x| \geq 2$.
  $\Rightarrow$ The length product decreases by a geometric factor.
- Quadratic complexity:
  computing x: $O(\log ||\mathbf{b}_2|| \cdot [1 + \log ||\mathbf{b}_2|| - \log ||\mathbf{b}_1||])$.
  $\Rightarrow O\left(\sum_{i=1}^{\tau} \log \|\mathbf{b}_2^i\| \cdot [1 + \log \|\mathbf{b}_2^i\| - \log \|\mathbf{b}_1^i\|]\right)$
  $= O\left(\log \|\mathbf{b}_2^0\| \cdot \sum_{i=1}^{\tau} [1 + \log \|\mathbf{b}_2^i\| - \log \|\mathbf{b}_2^{i+1}\|]\right)$
  $= O\left(\log \|\mathbf{b}_2\| \cdot [\tau + \log \|\mathbf{b}_2\| - \log \lambda_1(L)]\right).$

The Greedy Algorithm (1/2)



The Greedy Algorithm (2/2)

Name: $\mathsf{Greedy}(\mathbf{a}_1, \ldots, \mathbf{a}_d)$.

Input: A basis $[\mathbf{a}_1, \ldots, \mathbf{a}_d]$.

Output: A G-reduced basis of $L[\mathbf{a}_1, \ldots, \mathbf{a}_d]$.

1. If $d = 1$, return $[\mathbf{a}_1]$.
2. Repeat
3. Sort $(\mathbf{a}_1, \ldots, \mathbf{a}_d)$ by increasing lengths,
4. $[\mathbf{b}_1, \ldots, \mathbf{b}_{d-1}] := \mathsf{Greedy}(\mathbf{a}_1, \ldots, \mathbf{a}_{d-1})$,
5. Find a closest vector $\mathbf{c}$ to $\mathbf{a}_d$, in $L[\mathbf{b}_1, \ldots, \mathbf{b}_{d-1}]$,
6. $\mathbf{b}_d := \mathbf{a}_d - \mathbf{c}$,
7. Until $||\mathbf{b}_d|| \geq ||\mathbf{b}_{d-1}||$.
8. Return $[\mathbf{b}_1, \ldots, \mathbf{b}_d]$.

Termination and Correctness

- Termination: the length product decreases at each iteration.
- Correctness: equivalence up to dim 4 of G- and M-reductions.

- $[\mathbf{b}_1, \ldots, \mathbf{b}_d]$ is G-reduced
  $\Leftrightarrow \forall i, \forall x_1, \ldots, x_{i-1} \in \mathbb{Z}, ||\mathbf{b}_i + x_1 \mathbf{b}_1 + \ldots + x_{i-1} \mathbf{b}_{i-1}|| \geq ||\mathbf{b}_i||$
  $\Leftrightarrow \forall i, \mathrm{Proj}_{i-1} \mathbf{b}_i \in \mathrm{Vor}[\mathbf{b}_1, \ldots, \mathbf{b}_{i-1}]$.
- $[\mathbf{b}_1, \ldots, \mathbf{b}_d]$ is M-reduced iff $||x_1 \mathbf{b}_1 + \ldots + x_d \mathbf{b}_d|| \geq ||\mathbf{b}_i||$
  for all i and for all $x_1, \ldots, x_d \in \mathbb{Z}$ with $\gcd(x_i, \ldots, x_d) = 1$.

Minkowski Conditions

Let $d \leq 5$. A basis $[\mathbf{b}_1, \ldots, \mathbf{b}_d]$ is M-reduced iff $\forall i, \forall x_1, \ldots, x_d$ with $\gcd(x_i, \ldots, x_d) = 1$ and $|x_1|, \ldots, |x_d|$ is in the table below (up to any indices permutation), then $||x_1\mathbf{b}_1 + \ldots + x_d\mathbf{b}_d|| \geq ||\mathbf{b}_i||$.

1	1			
1	1	1		
1	1	1	1	
1	1	1	1	1
1	1	1	1	2

For example, with $d = 3$:

$||\mathbf{b}_3|| \geq ||\mathbf{b}_2|| \geq ||\mathbf{b}_1||$,

$||\mathbf{b}_2 \pm \mathbf{b}_1|| \geq ||\mathbf{b}_2||$,

$||\mathbf{b}_3 \pm \mathbf{b}_1|| \geq ||\mathbf{b}_3||$,

$||\mathbf{b}_3 \pm \mathbf{b}_2|| \geq ||\mathbf{b}_3||$,

$||\mathbf{b}_3 \pm \mathbf{b}_1 \pm \mathbf{b}_2|| \geq ||\mathbf{b}_3||$.

The Algorithm Fails in Dimension 5

$$\begin{bmatrix} 2 & 0 & 0 & 0 & 1 \\ 0 & 2 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 & 1 \\ 0 & 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 0 & \varepsilon \end{bmatrix}$$

G-reduced but not M-reduced for $\varepsilon \in \,]0,1[$.

Some Notations

- Beginning of the loop iteration: $[\mathbf{a}_1, \ldots, \mathbf{a}_d]$.
- After the recursive call: $[\mathbf{b}_1, \ldots, \mathbf{b}_{d-1}, \mathbf{a}_d]$.
- $\mathbf{c} = x_1\mathbf{a}_1 + \ldots + x_{d-1}\mathbf{a}_{d-1}$ a closest vector to $\mathbf{a}_d$.
- $\mathbf{b}_d = \mathbf{a}_d - \mathbf{c}$.
- π = rank of $\mathbf{b}_d$ once $(\mathbf{b}_1, \ldots, \mathbf{b}_d)$ is re-ordered (at the following loop iteration).

General Overview of the Complexity Analysis

- Linear number of loop iterations $\Leftarrow$ geometric decrease of the length product in any $O(1)$ consecutive iterations:
  - At least once every d loop iterations, $|x_{\pi_{i-1}}| \geq 2$.
  - $[\mathbf{a}_1, \ldots, \mathbf{a}_{d-1}]$ not quasi-reduced: geometric decrease at the previous loop iteration.
  - Obvious if $\mathbf{a}_\pi, \ldots, \mathbf{a}_d$ have not $\approx$ the same lengths.
  - Otherwise, we use the Gap Lemma: $\mathrm{Proj}_{d-1}\mathbf{a}_d$ is far from $\mathrm{Vor}[\mathbf{a}_1, \ldots, \mathbf{a}_{d-1}]$, thus $\mathbf{b}_d$ is far shorter than $\mathbf{a}_d$.

- Analysing precisely the cost of the CVP routine.
- Bounding cleverly the costs of the successive loop iterations.

What are the Difficult Points?

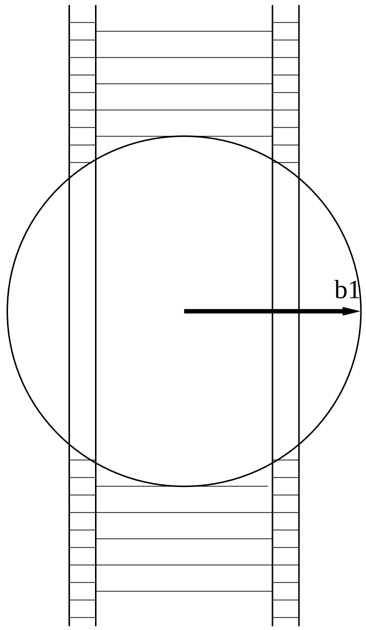
- Dealing with the non-determinism of the re-ordering.
- Defining what “quasi-reduced” means.
- Proving the Gap Lemma.
- Working around the fact that the Gap Lemma is partly wrong in dim 4.
- Bounding very tightly the cost of the CVP routine.

Sometimes we get a 2

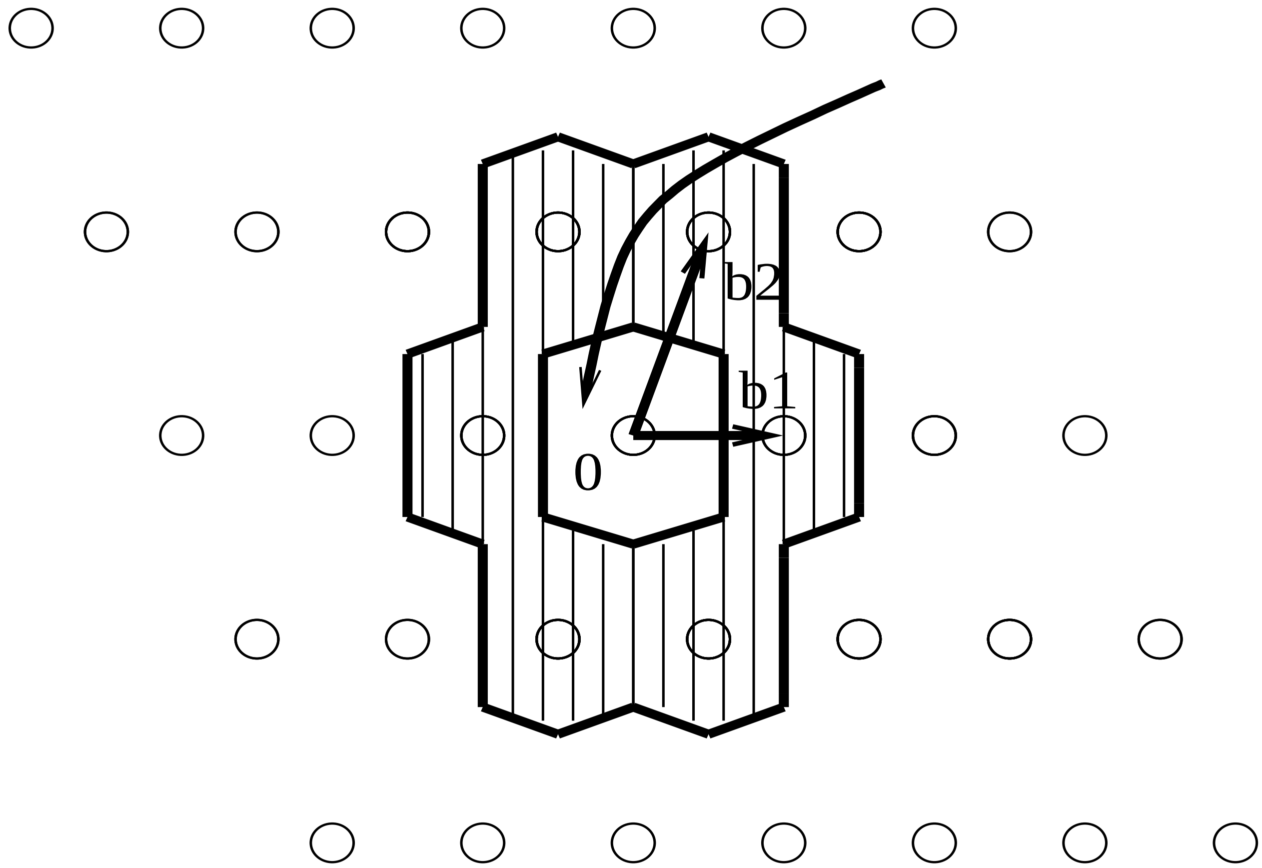
- Suppose that $d = 3$ and $\pi_{i-1} = 2$.
- $\mathbf{b}_3 = \mathbf{a}_3 - \mathbf{c} = \mathbf{a}_3 + x_1\mathbf{a}_1 + x_2\mathbf{a}_2$.
- Three cases:
  - $x_2 = 0$: $[\mathbf{a}_1, \mathbf{a}_3]$ is the "$[\mathbf{b}_1, \mathbf{b}_2]$" of the previous loop iteration, which is reduced. $\mathbf{b}_3 = \mathbf{a}_3$, last iteration.
  - $|x_2| = 1$: $\mathbf{b}_3 = \mathbf{a}_3 + x_1\mathbf{a}_1 \pm \mathbf{a}_2 = \pm\mathbf{a}_2 + \mathbf{a}_3 + x_1\mathbf{a}_1$. Because of the previous loop, $\mathbf{a}_2$ cannot be shortened by using $\mathbf{a}_3$ and $\mathbf{a}_1$. $||\mathbf{b}_3|| \geq ||\mathbf{a}_2|| \geq ||\mathbf{b}_2||$, and $\pi_i \geq \pi_{i-1} + 1$.
  - Otherwise, we get a 2.

Quasi-Reduced Bases

- $[\mathbf{b}_1, \ldots, \mathbf{b}_d]$ is G-reduced $\Leftrightarrow \forall i, \mathrm{Proj}_{i-1}\mathbf{b}_i \in \mathrm{Vor}[\mathbf{b}_1, \ldots, \mathbf{b}_{i-1}]$.
- Let $\varepsilon > 0$. $[\mathbf{b}_1, \ldots, \mathbf{b}_d]$ is ε-reduced $\Leftrightarrow \forall i, \mathrm{Proj}_{i-1}\mathbf{b}_i \in (1+\varepsilon)\mathrm{Vor}[\mathbf{b}_1, \ldots, \mathbf{b}_{i-1}]$.
- If $[\mathbf{a}_1, \ldots, \mathbf{a}_{d-1}]$ is not ε-reduced, then we had a geom. decrease at the previous loop iteration.
- Q: How properties on reduced bases can be extended to quasi-reduced bases?



The Gap Lemma (1/2)



The Gap Lemma (2/2)

- Let $2 \leq d \leq 4$. $\exists \varepsilon, D > 0$ s.t. the following holds.
- Let $[\mathbf{b}_1, \ldots, \mathbf{b}_{d-1}]$ an ε-reduced basis, $\mathbf{u} \in \mathrm{Vor}[\mathbf{b}_1, \ldots, \mathbf{b}_{d-1}]$ and $x_1, \ldots, x_{d-1}$ be integers.
- If $||\mathbf{b}_k|| \geq (1-\varepsilon)||\mathbf{b}_{d-1}||$ for some $k \leq d-1$, then:

$$||\mathbf{u}||^2 + D||\mathbf{b}_k||^2 \leq ||\mathbf{u} + \sum_{j=1}^{d-1} x_j \mathbf{b}_j||^2,$$

where $|x_k| \geq 2$, and if $d = 4$ the two other $|x_j|$'s are not both 1.

Open Problems

- Fast (quasi-linear time) version of the algorithm.
- What happens in dimension 5? and beyond?
- Can we use some of the tools of the proof anywhere else?