The unique-SVP World

Shai Halevi, IBM, July 2009

1. Ajtai-Dwork’97/07, Regev’03
   - PKE from worst-case uSVP
2. Lyubashvsky-Micciancio’09
   - Relations between worst-case uSVP, BDD, GapSVP

Many slides stolen from Oded Regev, denoted by ®
**f(n)-unique-SVP**

- **Promise:** the shortest vector $u$ is shorter by a factor of $f(n)$
- **Algorithm for $2^n$-unique SVP** [LLL82, Schnorr87]
- **Believed to be hard for any polynomial $n^c$**

1  $n^c$  $2^n$

believed hard  easy

$1 \geq f(n)$
**Ajtai-Dwork & Regev’03 PKEs**

- **Worst-case Search** u-SVP
  - Regev03: “Hensel lifting”
  - AD97: Geometric

- **Worst-case Decision** u-SVP

- **“Worst-case Distinguisher”** Wavy-vs-Uniform
  - Basic Intuition
  - Leftover hash lemma

- **AD97 PKE** bit-by-bit n-dimensional
  - Projecting to a line
  - Amortizing by adding dimensions

- **Regev03 PKE** bit-by-bit 1-dimensional

- **AD07 PKE** O(n)-bits n-dimensional

**Nearly-trivial worst-case/average-case reductions**
n-dimensional distributions

- Distinguish between the distributions:

- Wavy
  (In a random direction)

- Uniform
Given a lattice $L$, the dual lattice is

$$L^* = \{ x \mid \text{for all } y \in L, \langle x, y \rangle \in \mathbb{Z} \}$$
<table>
<thead>
<tr>
<th>Case 1</th>
<th></th>
<th>Case 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Diagram for Case 1" /></td>
<td><img src="image2.png" alt="Diagram for L" /></td>
<td><img src="image3.png" alt="Diagram for Case 2" /></td>
</tr>
</tbody>
</table>

**L* - the dual of L**

- Case 1: Diagram showing a sequence of points with a circle and a direction marked by a line with a length of $\frac{1}{n}$.
- Case 2: Diagram showing a more complex arrangement of points with a circle and a line with a length of $\frac{1}{\sqrt{n}}$. The diagram also includes a larger, more dense arrangement of points.
Reduction

- **Input:** a basis $B^*$ for $L^*$
- **Produce a distribution that is:**
  - Wavy if $L$ has unique shortest vector ($|u| \leq 1/n$)
  - Uniform (on $P(B^*)$) if $\lambda_1(L) > \sqrt{n}$
- **Choose a point from a Gaussian of radius $\sqrt{n}$, and reduce mod $P(B^*)$**
  - Conceptually, a “random $L^*$ point” with a $\text{Gaussian}(\sqrt{n})$ perturbation
<table>
<thead>
<tr>
<th>Case 2</th>
<th>Case 1</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Image of dots" /></td>
<td><img src="image2" alt="Image of dots" /></td>
</tr>
</tbody>
</table>

Creating the Distribution

$L^*$

$L^*$ + perturb
Analyzing the Distribution

- Theorem: (using [Banaszczyk’93])
  The distribution obtained above depends only on the points in L of distance $\sqrt{n}$ from the origin (up to an exponentially small error)

- Therefore,
  
  Case 1: Determined by multiples of $u$ →
  wavy on hyperplanes orthogonal to $u$

  Case 2: Determined by the origin →
  uniform
Proof of Theorem

- For a set $A$ in $\mathbb{R}^n$, define:
  \[ \rho(A) = \sum_{x \in A} e^{-\pi \|x\|^2} \]

- Poisson Summation Formula implies:
  \[ \forall y \in P(L^*), \quad \rho(y - L^*) = d(L) \cdot \sum_{x \in L} e^{2\pi i <x, y>} \rho(\{x\}) \]

- Banaszczyk’s theorem:
  For any lattice $L$,
  \[ \rho(L - \sqrt{n}B_n) < 2^{-\Omega(n)} \rho(L \cap \sqrt{n}B_n) \]
Proof of Theorem (cont.)

In Case 2, the distribution obtained is very close to uniform:

\[ \forall y \in P(L^*), \quad \rho(y - L^*) = d(L) \cdot \sum_{x \in L} e^{2\pi i \langle x, y \rangle} \rho(\{x\}) = \]

\[ d(L) \cdot \left(1 + \sum_{x \in L \setminus \{0\}} e^{2\pi i \langle x, y \rangle} \rho(\{x\})\right) \approx d(L) \]

Because:

\[ \left| \sum_{x \in L \setminus \{0\}} e^{2\pi i \langle x, y \rangle} \rho(\{x\}) \right| < \sum_{x \in L \setminus \{0\}} \rho(\{x\}) = \]

\[ \rho(L \setminus \{0\}) = \rho(L - \sqrt{n}B_n) < 2^{-\Omega(n)} \rho(L \cap \sqrt{n}B_n) = 2^{-\Omega(n)} \]
Ajtai-Dwork & Regev’03 PKEs

- Worst-case Search u-SVP
- AD97: Geometric
- Regev03: “Hensel lifting”
- “Worst-case Distinguisher” Wavy-vs-Uniform
- Basic Intuition
- Worst-case Decision u-SVP
Distinguish $\rightarrow$ Search, AD97

- Reminder: $L^*$ lives in hyperplanes

- We want to identify $u$
  - Using an oracle that distinguishes wavy distributions from uniform in $P(B^*)$
The plan

1. Use the oracle to distinguish points close to $H_0$ from points close to $H_{\pm 1}$
2. Then grow very long vectors that are rather close to $H_0$
3. This gives a very good approximation for $u$, then we use it to find $u$ exactly
Distinguishing $H_0$ from $H_{\pm 1}$

Input: basis $B^*$ for $L^*$, $\sim$ length of $u$, point $x$

- And access to wavy/uniform distinguisher

Decision: Is $x$ $1/\text{poly}(n)$ close to $H_0$ or to $H_{\pm 1}$?

- Choose $y$ from a wavy distribution near $L^*$
  - $y = \text{Gaussian}(\sigma)^*$ with $\sigma < 1/2|u|$
- Pick $\alpha \in \mathbb{R}[0,1]$, set $z = \alpha x + y \mod P(B^*)$
- Ask oracle if $z$ is drawn from wavy or uniform distribution

* Gaussian($\sigma$): variance $\sigma^2$ in each coordinate
Distinguishing $H_0$ from $H_{\pm 1}$ (cont.)

Case 1: $x$ close to $H_0$

- $\alpha x$ also close to $H_0$
- $\alpha x + y \mod P(B^*)$ close to $L^*$, wavy
Distinguishing $H_0$ from $H_{\pm 1}$ (cont.)

Case 2: $x$ close to $H_{\pm 1}$

- $\alpha x$ “in the middle” between $H_0$ and $H_{\pm 1}$
  - Nearly uniform component in the $u$ direction
- $\alpha x + y \mod P(B^*)$ nearly uniform in $P(B^*)$
Distinguishing $H_0$ from $H_{\pm 1}$ (cont.)

- Repeat poly(n) times, take majority
  - Boost the advantage to near-certainty
- Below we assume a “perfect distinguisher”
  - Close to $H_0 \implies$ always says NO
  - Close to $H_{\pm 1} \implies$ always says YES
  - Otherwise, there are no guarantees
    - Except halting in polynomial time
Growing Large Vectors

- Start from some $x_0$ between $H_{-1}$ and $H_{+1}$
  - e.g. a random vector of length $1/|u|$
- In each step, choose $x_i$ s.t.
  - $|x_i| \sim 2|x_{i-1}|$
  - $x_i$ is somewhere between $H_{-1}$ and $H_{+1}$
- Keep going for $\text{poly}(n)$ steps
- Result is $x^*$ between $H_{\pm 1}$ with $|x^*| = N/|u|$
  - Very large $N$, e.g., $N = 2^{n^2}$
From $x_{i-1}$ to $x_i$

Try poly(n) many candidates:

- Candidate $w = 2x_{i-1} + \text{Gaussian}(1/|u|)$
- For $j = 1, \ldots, m = \text{poly}(n)$
  - $w_j = j/m \cdot w$
  - Check if $w_j$ is near $H_0$ or near $H_{\pm 1}$
- If none of the $w_j$’s is near $H_{\pm 1}$ then accept $w$ and set $x_i = w$
- Else try another candidate
From $x_{i-1}$ to $x_i$: Analysis

- $x_{i-1}$ between $H_{\pm 1} \rightarrow w$ is between $H_{\pm n}$
  - Except with exponentially small probability
- $w$ is NOT between $H_{\pm 1} \rightarrow$ some $w_j$ near $H_{\pm 1}$
  - So $w$ will be rejected
- So if we make progress, we know that we are on the right track
From $x_{i-1}$ to $x_i$: Analysis (cont.)

- With probability $1/\text{poly}(n)$, $w$ is close to $H_0$
  - The component in the $u$ direction is Gaussian with mean $< 2/|u|$ and variance $1/|u|^2$
From $x_{i-1}$ to $x_i$: Analysis (cont.)

- With probability $1/\text{poly}$, $w$ is close to $H_0$
  - The component in the $u$ direction is Gaussian
    with mean $< 2/|u|$ and standard deviation $1/|u|$ 

- $w$ is close to $H_0$, all $w_j$’s are close to $H_0$
  - So $w$ will be accepted 

- After polynomially many candidates, we will make progress whp
Finding $u$

- **Find $n-1$ $x^*$’s**
  - $x^*_{t+1}$ is chosen orthogonal to $x^*_1, \ldots, x^*_t$
  - By choosing the Gaussians in that subspace

- **Compute $u' \perp \{x^*_1, \ldots, x^*_{n-1}\}$, with $|u'| = 1$**
  - $u'$ is exponentially close to $u/|u|$
    - $u/|u| = (u' + e)$, $|e| = 1/N$
    - Can make $N \gg 2^n$ (e.g., $N = 2^{n^2}$)

- **Diophantine approximation to solve for $u$**
Ajtai-Dwork & Regev’03 PKEs

Worst-case Search u-SVP

Regev03: “Hensel lifting”

“Worst-case Distinguisher” Wavy-vs-Uniform

Worst-case/average-case + leftover hash lemma

AD97 PKE bit-by-bit n-dimensional

AD97: Geometric

Basic Intuition

(slide 47)
Average-case Distinguisher

- Intuition: lattice only matters via the direction of $u$
- Security parameter $n$, another parameter $N$
- A random $u$ in $n$-dim. unit sphere defines $\mathcal{D}_u(N)$
  - $\chi = \text{disceret-Gaussian}(N)$ in one dimension
    - Defines a vector $x=\chi \cdot u <u,u>$, namely $x \parallel u$ and $<x,u>=\chi$
  - $y = \text{Gaussian}(N)$ in the other $n-1$ dimensions
  - $e = \text{Gaussian}(n^{-4})$ in all $n$ dimensions
- Output $x+y+e$
Worst-case/average-case (cont.)

Thm: Distinguishing $\mathcal{D}_u(N)$ from Uniform

$\rightarrow$ Distinguishing Wavy$_{B^*}$ from Uniform$_{B^*}$ for all $B^*$

- When you know $\lambda_1(L(B))$ upto $(1+1/poly(n))$-factor
- For parameter $N = 2^{\Omega(N)}$

Pf: Given $B^*$, scale it s.t. $\lambda_1(L(B)) \in [1,1+1/poly)$

- Also apply random rotation

- Given samples $x$ (from Uniform$_{B^*}$ / Wavy$_{B^*}$)
  - Sample $y=$discrete-Gaussian$_{B^*}(N)$
    - Can do this for large enough $N$
  - Output $z=x+y$

- “Clearly” $z$ is close to $\mathcal{G}(N) / \mathcal{D}_u(N)$ respectively
The AD97 Cryptosystem

- **Secret key:** a random \( u \in \) unit sphere

- **Public key:** \( n+m+1 \) vectors (\( m=8n \log n \))
  - \( b_1, \ldots, b_n \leftarrow \mathcal{D}_u(2^n), \quad v_0, v_1, \ldots, v_m \leftarrow \mathcal{D}_u(n2^n) \)
    - So \( \langle b_i, u \rangle, \langle v_i, u \rangle \) ~ integer
    - We insist on \( \langle v_0, u \rangle \) ~ odd integer

- **Will use** \( P(b_1, \ldots, b_n) \) for encryption
  - Need \( P(b_1, \ldots, b_n) \) with “width” > \( 2^n/n \)
The AD97 Cryptosystem (cont.)

Encryption(\(\sigma\)):
- \(c' \leftarrow \text{random-subset-sum}(v_1,\ldots,v_m) + \sigma v_0/2\)
- output \(c = (c' + \text{Gaussian}(n^{-4})) \mod P(B)\)

Decryption(\(c\)):
- If \(<u,c>\) is closer than \(\frac{1}{4}\) to integer say 0,
  else say 1

Correctness due to \(<b_i,u>,<v_j,u>\sim\text{integer}\)
  - and width of \(P(B)\)
AD97 Security

- The $b_i$’s, $v_i$’s chosen from $\mathcal{D}_u$(something)
- By hardness assumption, can’t distinguish from $\mathcal{G}_u$(something)
- Claim: if they were from $\mathcal{G}_u$(something), $c$ would have no information on the bit $\sigma$
  - Proven by leftover hash lemma + smoothing
- Note: $v_i$’s has variance $n^2$ larger than $b_i$’s
  - In the $\mathcal{G}_u$ case $v_i \mod P(B)$ is nearly uniform
Partition \( P(B) \) to \( q^n \) cells, \( q \sim n^7 \)

For each point \( v_i \), consider the cell where it lies
- \( r_i \) is the corner of that cell

\[ \sum_S v_i \mod P(B) = \sum_S r_i \mod P(B) + n^{-5} \text{ “error”} \]
- \( S \) is our random subset

\[ \sum_S r_i \mod P(B) \] is a nearly-random cell
- We’ll show this using leftover hash

The Gaussian \( (n^{-4}) \) in \( c \) drowns the error term
Leftover Hashing

- Consider hash function $H_R : \{0,1\}^m \rightarrow [q]^n$
  - The key is $R=[r_1,\ldots,r_m] \in [q]^{n \times m}$
  - The input is a bit vector $b=[\sigma_1,\ldots,\sigma_m]^T \in \{0,1\}^m$
- $H_R(b) = Rb \mod q$
- $H$ is “pairwise independent” (well, almost..)
  - Yay, let’s use the leftover hash lemma
- $\langle R,H_R(b) \rangle$, $\langle R,U \rangle$ statistically close
  - For random $R \in [q]^{n \times m}$, $b \in \{0,1\}^m$, $U \in [q]^n$
  - Assuming $m \gg n \log q$
AD97 Security (cont.)

- We proved $\sum_S r_i \mod P(B)$ is nearly-random
- Recall:
  - $c_0 = \sum_S r_i + \text{error}(n^{-5}) + \text{Gaussian}(n^{-4}) \mod P(B)$
- For any $x$ and error $e$, $|e| \sim n^{-5}$, the distr. $x + e + \text{Gaussian}(n^{-5})$, $x + \text{Gaussian}(n^{-4})$ are statistically close
- So $c_0 \sim \sum_S r_i + \text{Gaussian}(n^{-3}) \mod P(B)$
  - Which is close to uniform in $P(B)$
  - Also $c_1 = c_0 + v_0/2 \mod P(B)$ close to uniform
Ajtai-Dwork & Regev’03 PKEs

- Worst-case Search u-SVP
  - AD97: Geometric
- Average-case Decision Wavy-vs-Uniform
  - Leftover hash lemma
- Worst-case Decision u-SVP
  - Basic Intuition
- Regev03: “Hensel lifting”
- AD97 PKE bit-by-bit n-dimensional
  - Amortizing by adding dimensions
- Regev03 PKE bit-by-bit 1-dimensional
  - Not today
- AD07 PKE O(n)-bits n-dimensional

(slides 60) Projecting to a line

Not today
u-SVP vs. BDD vs. GAP-SVP

Lyubashevsky-Micciancio, CRYPTO 2009

Good old-fashion worst-case reductions
- Mostly Cook reductions (one Karp reduction)
Reminder: uSVP and BDD

**uSVP**$_\gamma$: $\gamma$-unique shortest vector problem
- **Input:** a basis $B = (b_1, \ldots, b_n)$
- **Promise:** $\lambda_1(L(B)) < \gamma \lambda_2(L(B))$
- **Task:** find shortest nonzero vector in $L(B)$

**BDD**$_{1/\gamma}$: $1/\gamma$-bounded distance decoding
- **Input:** a basis $B = (b_1, \ldots, b_n)$, a point $t$
- **Promise:** $\text{dist}(t, L(B)) < \lambda_1(L(B)) / \gamma$
- **Task:** find closest vector to $t$ in $L(B)$
BDD$_{1/\gamma}$ $\leq$ uSVP$_{\gamma/2}$

- **Input:** a basis $B = (b_1, \ldots, b_n)$, a point $t$
  - Assume that we know $\mu = \text{dist}(t, L(B))$

- **Let $B' = \begin{pmatrix} b_1 & \ldots & b_n & t \\ 0 & 0 & \mu \end{pmatrix}$**
  - Let $v \in L(B)$ be the closest to $t$, $|t-v|=\mu$
  - Will show that the vector $[(t-v) \ \mu]^{T}$ is the $\gamma/2$-unique shortest vector in $L(B')$
  - So uSVP$_{\gamma/2}(B')$ will return it

- **The size of $v' = [(t-v) \ \mu]^{T}$ is $(\mu^2+\mu^2)^{1/2} = \sqrt{2}\times\mu$**

Can get by with a good approximation for $\mu$
Every $w' \in L(B')$ looks like $w' = [\beta t - w \beta \mu]^T$

- For some integer $\beta$ and some $w \in L(B)$
- Write $\beta t - w = (\beta v - w) - \beta (v - t)$
- $\beta v - w \in L(B)$, nonzero if $w'$ isn’t a multiple of $v'$
- So $|\beta v - w| \geq \lambda_1$, also recall $|v - t| = \mu \leq \lambda_1 / \gamma$

$|\beta t - w| \geq |\beta v - w| - \beta |v - t| \geq \lambda_1 - \beta \mu$

$|w'|^2 \geq (\lambda_1 - \beta \mu)^2 + (\beta \mu)^2 \geq \inf_{\beta \in \mathbb{R}}[(\lambda_1 - \beta \mu)^2 + (\beta \mu)^2]$

$= (\lambda_1)^2 / 2 \geq (\gamma \mu)^2 / 2$

So for any $w' \in L(B')$, not a multiple of $v'$, we have $|w'| \geq \mu \gamma / 2 = |v'| \times \gamma / 2$
\[
\text{uSVP}_\gamma \leq \text{BDD}_{1/\gamma}
\]

- **Input**: a basis \( B = (b_1, b_2, \ldots, b_n) \)
  - Let \( \rho \) be a prime, \( \rho \geq \gamma \)
- For \( i=1,2,\ldots,n, \ j=1,2,\ldots,p-1 \)
  - \( B_i = (b_1, b_2, \ldots, \rho \times b_i, \ldots, b_n) \), \( t_{ij} = j \times b_i \)
  - Let \( v_{ij} = \text{BDD}_{1/\gamma}(B_i, T_{ij}) \), \( w_{ij} = v_{ij} - t_{ij} \)
- **Output** the smallest nonzero \( w_{ij} \) in \( L(B) \)
Let $u$ be shortest nonzero vector in $L(B)$
- \( u = \sum \xi_i b_i \), at least one $\xi_i$ isn’t divisible by $\rho$
  (otherwise $u/\rho$ would also be in $L(B)$)
- Let $j = -\xi_i \mod \rho$, $j \in \{1, 2, \ldots, \rho-1\}$

We will prove that for these $i, j$
- $\lambda_1(L(B_i)) > \gamma \lambda_1(L(B))$
- $\text{dist}(t_{ij}, L(B_i)) \leq \lambda_1(L(B))$
The smallest multiple of $u$ in $L(B_i)$ is $\rho u$

- $|\rho u| = \rho \lambda_1(L(B)) \geq \gamma \lambda_1(L(B))$
- Any other vector in $L(B_i) \subseteq L(B)$ is longer than $\gamma \lambda_1(L(B))$ (since $L(B)$ is $\gamma$-unique)

$\lambda_1(L(B_i)) \geq \gamma \lambda_1(L(B))$

$t_{ij} + u = jb_i + \sum \tilde{\xi}_m b_m = (j+\tilde{\xi}_i)b_i + \sum_{m \neq i} \tilde{\xi}_m b_m \in L(B_i)$

$\text{dist}(t_{ij}, L(B_i)) \leq \lambda_1(L(B_i))$

$(B_i, t_{ij})$ satisfies the promise of $\text{BDD}_{1/\gamma}$

$v_{ij} = \text{BDD}_{1/\gamma}(B_i, t_{ij})$ is closest to $t_{ij}$ in $L(B_i)$
- $w_{ij} = v_{ij} - t_{ij} \in L(B)$, since $t_{ij} \in L(B)$ and $v_{ij} \in L(B_i) \subseteq L(B)$
- $|w_{ij}| = \lambda_1(L(B))$
Reminder: GapSVP

- **GapSVP$_\gamma$**: decision version of approx$_\gamma$-SVP
  - Input: Basis $B$, number $\delta$
  - Promise: either $\lambda_1(L(B)) \leq \delta$ or $\lambda_1(L(B)) > \gamma \delta$
  - Task: decide which is the case

- The reduction $u$SVP$_\gamma \leq$ GapSVP$_\gamma$ is the same as Regev’s Decision-to-Search $u$SVP reduction
GapSVP $\gamma \sqrt{n \log n} \leq \text{BDD}_{1/\gamma}$

- **Inputs:** Basis $B=(b_1,\ldots,b_n)$, number $\delta$
- **Repeat** poly$(n)$ times
  - Choose a random $s_i$ of length $\leq \delta \sqrt{n \log n}$
  - Set $t_i = s_i \mod B$, run $v_i = \text{BDD}_{1/\gamma}(B,t_i)$
- **Answer YES** if $\exists i$ s.t. $v \neq t_i - s_i$, else **NO**

Need will show:

- $\lambda_1(L(B)) > \gamma \delta \sqrt{n \log n} \Rightarrow v = t_i - s_i$ always
- $\lambda_1(L(B)) \leq \delta \Rightarrow v \neq t_i - s_i$ with probability $\sim 1/2$
Case 1: $\lambda_1(L(B)) > \sqrt[n]{n \log n} \cdot \delta$

- Recall: $|s_i| \leq \delta \sqrt[n]{n \log n}$, $t_i = s_i \mod B$
  - $t_i$ is $\leq \delta \sqrt[n]{n \log n}$ away from $v_i = t_i - s_i \in L(B)$
  - $(B, t_i)$ satisfies the promise of $BDD_{1/\gamma}$
  - $BDD_{1/\gamma}(B, t_i)$ will return some vector in $L(B)$

- Any other $L(B)$ point has distance from $t_i$ at least $\lambda_1(L(B)) - \delta \sqrt[n]{n \log n} > (\gamma - 1) \delta \sqrt[n]{n \log n}$
  - $v_i$ is only answer that $BDD_{1/\gamma}(B, t_i)$ can return
Case 2: $\lambda_1(L(B)) \leq \delta$

- Let $u$ be shortest nonzero in $L(B)$, $|u| = \lambda_1$
- $s_i$ is random in $\text{Ball}(\delta \sqrt{n \log n})$
- With high probability $s_i \pm u$ also in ball
  - $t_i = s_i \mod B$ could just as well be chosen as $t_i = (s_i + u) \mod B$
  - Whatever $\text{BDD}_{1/\gamma}(B,t)$ returns it differs from $t_i - s_i$ w.p. $\geq 1/2$
Backup Slides

1. Regev’s Decision-to-Search uSVP
2. Regev’s dimension reduction
3. Diophantine Approximation
uSVP Decision $\rightarrow$ Search

- Search-uSVP
- Decision mod-p problem
- Decision-uSVP
Reduction from: Decision mod-p

- Given a basis \((v_1 \ldots v_n)\) for \(n^{1.5}\)-unique lattice, and a prime \(p > n^{1.5}\)
- Assume the shortest vector is:
  \[ u = a_1 v_1 + a_2 v_2 + \ldots + a_n v_n \]
- Decide whether \(a_1\) is divisible by \(p\)
Reduction to:
Decision uSVP

- Given a lattice, distinguish between:
  - Case 1. Shortest vector is of length $1/n$ and all non-parallel vectors are of length more than $\sqrt{n}$
  - Case 2. Shortest vector is of length more than $\sqrt{n}$
The reduction

- Input: a basis \((v_1, \ldots, v_n)\) of a \(n^{1.5}\) unique lattice
- Scale the lattice so that the shortest vector is of length \(1/n\)
- Replace \(v_1\) by \(pv_1\). Let \(M\) be the resulting lattice
- If \(p \mid a_1\) then \(M\) has shortest vector \(1/n\) and all non-parallel vectors more than \(\sqrt{n}\)
- If \(p \nmid a_1\) then \(M\) has shortest vector more than \(\sqrt{n}\)
The input lattice $L$
The lattice $M$

- The lattice $M$ is spanned by $pv_1, v_2, \ldots, v_n$.
- If $p|a_1$, then $u = (a_1/p) \cdot pv_1 + a_2 v_2 + \ldots + a_n v_n \in M$.
The lattice $M$ is spanned by $pv_1, v_2, \ldots, v_n$:

- If $p \nmid a_1$, then $u \notin M$: 

\[ \sqrt{n} \]
Reduction from:  
Decision mod-p

- Given a basis \((v_1...v_n)\) for \(n^{1.5}\)-unique lattice, and a prime \(p>n^{1.5}\)
- Assume the shortest vector is:
  \[ u = a_1v_1+a_2v_2+...+a_nv_n \]
- Decide whether \(a_1\) is divisible by \(p\)
The Reduction

- Idea: decrease the coefficients of the shortest vector

- If we find out that $p|a_1$ then we can replace the basis with $pv_1, v_2, ..., v_n$.

- $u$ is still in the new lattice:

$$u = \frac{a_1}{p} \cdot pv_1 + a_2 v_2 + ... + a_n v_n$$

- The same can be done whenever $p|a_i$ for some $i$
The Reduction

- But what if $p \nmid a_i$ for all $i$?
- Consider the basis $v_1, v_2 - v_1, v_3, ..., v_n$
- The shortest vector is

$$u = (a_1 + a_2)v_1 + a_2(v_2 - v_1) + a_3v_3 + ... + a_nv_n$$
- The first coefficient is $a_1 + a_2$
- Similarly, we can set it to

$$a_1 - bp/2c a_2, ..., a_1 - a_2, a_1, a_1 + a_2, ..., a_1 + bp/2c a_2$$
- One of them is divisible by $p$, so we choose it and continue
The Reduction

- Repeating this process decreases the coefficients of $u$ are by a factor of $p$ at a time
  - The basis that we started from had coefficients $\leq 2^{2n}$
  - The coefficients are integers
- After $\leq 2n^2$ steps, all the coefficients but one must be zero
- The last vector standing must be $\pm u$
Regev’s dimension reduction
Reducing from n to 1-dimension

- Distinguish between the 1-dimensional distributions:

  Uniform:

  Wavy:
Reducing from n to 1-dimension

- First attempt: sample and project to a line
Reducing from n to 1-dimension

- But then we lose the wavy structure!
- We should project only from points very close to the line
The solution

- Use the periodicity of the distribution
- Project on a ‘dense line’:
The solution
The solution

- We choose the line that connects the origin to $e_1 + Ke_2 + K^2 e_3 + ... + K^{n-1} e_n$ where $K$ is large enough.

- The distance between hyperplanes is $n$.
- The sides are of length $2^n$.
- Therefore, we choose $K = 2^{O(n)}$.
- Hence, $d < O(K^n) = 2^{O(n^2)}$.
Worst-case vs. Average-case

- So far: a problem that is hard in the worst-case: distinguish between uniform and $d,\gamma$-wavy distributions for all integers $d < 2^{(n^2)}$
- For cryptographic applications, we would like to have a problem that is hard on the average: distinguish between uniform and $d,\gamma$-wavy distributions for a non-negligible fraction of $d$ in $[2^{(n^2)}, 2 \cdot 2^{(n^2)}]$
Compressing

- The following procedure transforms $d,γ$-wavy into $2d,γ$-wavy for all integer $d$:
  - Sample $a$ from the distribution
  - Return either $a/2$ or $(a+R)/2$ with probability $\frac{1}{2}$

- In general, for any real $\alpha \geq 1$, we can compress $d,γ$-wavy into $\alpha d,γ$-wavy

- Notice that compressing preserves the uniform distribution

- We show a reduction from worst-case to average-case
Assume there exists a distinguisher between uniform and $\gamma$-wavy distribution for some non-negligible fraction of $d$ in $[2^n, 2 \cdot 2^n)$]

- Given either a uniform or a $d, \gamma$-wavy distribution for some integer $d < 2^n$ repeat the following:
  - Choose $\alpha$ in $\{1, \ldots, 2 \cdot 2^n\}$ according to a certain distribution
  - Compress the distribution by $\alpha$
  - Check the distinguisher’s acceptance probability

- If for some $\alpha$ the acceptance probability differs from that of uniform sequences, return ‘wavy’; otherwise, return ‘uniform’
Reduction

- Distribution is uniform:
  - After compression it is still uniform
  - Hence, the distinguisher’s acceptance probability equals that of uniform sequences for all $\alpha$

- Distribution is $d,y$-wavy:
  - After compression it is in the good range with some probability
  - Hence, for some $\alpha$, the distinguisher’s acceptance probability differs from that of uniform sequences
Diophantine Approximation
Solving for \( u \)
(from slide 24)

- **Recall:** We have \( B=(b_1,\ldots,b_n) \) and \( u' \)
  - Shortest vector \( u \in L(B) \) is \( u = \sum \mu_i b_i, \ |\mu_i| < 2^n \)
    - Because the basis \( B \) is LLL reduced
  - \( u' \) is very very close to \( u/|u| \)
    - \( u/|u| = (u' + e), \ |e| = 1/N, \ N \gg 2^n \) (e.g., \( N=2^{n^2} \))

- **Express** \( u' = \sum \xi_i b_i \) (\( \xi_i \)'s are reals)

- **Set** \( v_i = \xi_i/\xi_n \) for \( i=1,\ldots,n-1 \)
  - \( v_i \) very very close to \( \mu_i/\mu_n \) (\( v_i \cdot \mu_n = \mu_i + O(2^n/N) \))
Diophantine Approximation

- Look for $\mu_n < 2^n$ s.t. for all $i$, $\nu_i \cdot \mu_n$ is $2^n/N$ away from an integer (for $N = 2^{n^2}$)

- $z$ is the unique shortest in $L(M)$ by a factor $\sim N/2^n$

- Use LLL to find it

- Compute the $\mu_i$'s and $u$
Why is \( z \) unique-shortest?

- Assume we have another short vector \( y \in L(M) \)
  - \( \mu_n \) not much larger than \( 2^n \), also the other \( \mu_i \)'s

- Every small \( y \in L(M) \) corresponds to \( v \in L(B) \) such that \( v/|v| \) very very close to \( u' \)
  - So also \( v/|v| \) very very close to \( u/|u| \) (\( \sim 2^n/N \))
  - Smallish coefficient \( \Rightarrow v \) not too long (\( \sim 2^{2n} \))

\( \Rightarrow v \) very close to its projection on \( u \) (\( \sim 2^{3n}/N \))

\( \Rightarrow \exists \chi \) s.t. \( (v-\chi u) \in L(B) \) is short
  - Of length \( \leq 2^{3n}/N + \lambda_1/2 < \lambda_1 \)

\( \Rightarrow v \) must be a multiple of \( u \)