Homomorphic Encryption and Lattices, Spring 2011	Instructor: Shai Halevi

Lecture 6: Lattice Trapdoor Constructions

April 7, 2011 — Scribe: Nir Bitansky

This lecture is based on the trapdoor constructions due to Ajtai [Ajt99], Alwen-Peikert[AP11], and Micciancio-Peikert [MP11].

In previous lectures, we have seen that, given a random matrix $A \in_R \mathbb{Z}_q^{n\times m}$ (with $q \geq \text{poly}(n)$ and $m \geq n \log q$), finding a short vector $\vec{v}$ such that $A\vec{v} = 0 \pmod q$ is at least as hard as obtaining a good SIVP approximation algorithm. (Where short means of size $O(\sqrt{m})$ and good means up to poly factors.)

We would like to generate A together with a short basis S for the lattice

$$\Lambda_q^\perp(A) \stackrel{\text{def}}{=} \{\vec{x} \in \mathbb{Z}^m : A\vec{x} = 0 \pmod q\}$$

Such a short basis can then be used to construct various cryptographic schemes, such as signatures, encryption, identity-based encryption and more.

We first note that $\det \Lambda_q^\perp(A) \leq q^n$.

Proof sketch. For any $\vec{u} \in [q-1]^n$ Consider the co-set

$$\vec{u} + \Lambda_q^\perp(A) = \{\vec{x} \in \mathbb{Z}^m : A\vec{x} = \vec{u} \pmod q\}$$

Then, $\det \Lambda_q^\perp(A)$ is the number of such distinct co-sets, which is at most q^n (and exactly q^n if A is of full rank). □

Therefore, by Minkowski, there exist vectors in $\Lambda_q^\perp(A)$ of size at most $\sqrt{m}q^{\frac{n}{m}}$. Our goal is to obtain a short basis $S \in \mathbb{Z}^{m\times m}$, where **all** vectors are of size $O(\sqrt{m}q^{\frac{n}{m}})$. We would also like m to be as small as possible, preferably $O(n \log q)$.

Easy exercise: Generate A with a single short vector $\vec{v} \in \Lambda_q^\perp(A)$. For this purpose, we can simply choose a random short vector $\vec{v} \in \{0,1\}^m$, and then choose a random A such that $A\vec{v} = 0 \pmod q$. Equivalently, choose the first $m-1$ columns of A at random, and the last column to be a random subset sum of the first columns. By the left over hash lemma (LOHL), A is statistically close to random, so long that $m > 3n \log q$.

Still easy: Generate A with t short vectors $\vec{v}_1, \ldots, \vec{v}_t \in \Lambda_q^\perp(A)$. Choose a random $A_1 \in_R \mathbb{Z}^{n\times m_1}$, where $m_1 = m - t$. Then choose $A_2 = -A_1 R$, where $R \in_R \{0,1\}^{m_1\times t}$. By LOHL, A is still statistically close to random, so long that $m_1 > 3n \log q$.

In general, using this naive method, we will always be $\Omega(n \log q)$ vectors short.

Starting with $A_1 \in \mathbb{Z}^{n\times m_1}$, can we add a single dimension and obtain two short vectors? This is actually almost as hard as finding a short vector for the initial A_1. Indeed, assume we add $\vec{a}$, and obtain short $(\vec{u}_1, \vec{u}_2) = ((\vec{v}_1, \gamma_1), (\vec{v}_2, \gamma_2))$ such that

$$\left(A_1 \mid \vec{a} \right) \begin{pmatrix} \vec{v}_1 & \vec{v}_2 \\ \gamma_1 & \gamma_2 \end{pmatrix} = 0 \pmod q$$

Then, $A_1(\gamma_2\vec{v}_1 - \gamma_1\vec{v}_2) = 0 \pmod q$, and the vector $\gamma_2\vec{v}_1 - \gamma_1\vec{v}_2$ is short and non-zero (since $\vec{u}_1, \vec{u}_2$ are independent). This still does not mean that we can not extend A_1 to obtain a short basis; namely, it is possible that if we add t dimensions we might obtain even more than t short vectors.

The Alwen-Peikert Construction

Let $m_1 + m_2 = m$. As a first step, let us try to extend a given $A_1 \in \mathbb{Z}^{n\times m_1}$ to $\left(A_1 \mid A_2 \right) \in \mathbb{Z}^{n\times m}$ together with a short basis $S \in \mathbb{Z}^{m\times m}$, allowing $A_2 \in \mathbb{Z}^{n\times m_2}$ not to be random. We require that

$$\left(A_1 \mid A_2 \right) \left(\begin{array}{c|c} V & W \\ \hline U & P \end{array} \right) = 0 \pmod q$$

For now we shall work with $W = 0$. After seeing that $U = I$ does not suffice, we will slightly augment the choice of U, while keeping it invertible. In what follows all equalities are done modulo q.

To obtain $A_1V + A_2U = 0$ we need $A_2 = -A_1VU^{-1}$. Let $G = VU^{-1}$. To obtain $A_1W + A_2P = 0$ we need $-A_1GP = 0$. Let $H = GP$. We wish to obtain:

$$S = \left(\begin{array}{c|c} GU & 0 \\ \hline U & P \end{array} \right)$$

such that U, GU, P are small (i.e., with small entries) and $H = GP \in \Lambda_q^{\perp}(A_1)$. Since we can not find short vectors in $\Lambda_q^{\perp}(A_1)$, H will be large. Adding the fact that P should be small, we deduce that G must also be large. That is, we are interested in finding small U and large G, such that GU is small.

First attempt: Consider

$$U = \begin{pmatrix} 1 & -1 & & \\ & \ddots & \ddots & \\ & & \ddots & -1 \\ & & & 1 \end{pmatrix}$$

then

$$\left(\vec{g}_1 \mid \dots \mid \vec{g}_t \right) U = \left(\vec{g}_1 \mid \vec{g}_2 - \vec{g}_1 \mid \dots \mid \vec{g}_t - \vec{g}_{t-1} \right)$$

This is not good enough since any column of G is a subset sum of columns in GU, implying that $\|G\|_\infty \le t\|GU\|_\infty$, and hence GU has large entries.

Second attempt: Consider

$$U = \begin{pmatrix} 1 & -2 & & \\ & \ddots & \ddots & \\ & & \ddots & -2 \\ & & & 1 \end{pmatrix}$$

then

$$\left(\vec{g}_1 \mid \dots \mid \vec{g}_t \right) U = \left(\vec{g}_1 \mid \vec{g}_2 - 2\vec{g}_1 \mid \dots \mid \vec{g}_t - 2\vec{g}_{t-1} \right)$$

Now we can have $\|\vec{g}_{i+1}\|_\infty \approx 2\|\vec{g}_i\|_\infty$ and GU can still potentially be small. Our final U will be based on the above. Let us for now denote by T_ℓ a matrix such as the above of dimension $\ell \times \ell$. For a given vector $\vec{h}$, let $\ell = \log \|\vec{h}\|_\infty$ (the maximum bit size of entries in $\vec{h}$). We define:

$$G[\vec{h}] \stackrel{\text{def}}{=} \left(\left\lfloor \frac{\vec{h}}{2^{\ell-1}} \right\rfloor \quad \dots \quad \left\lfloor \frac{\vec{h}}{4} \right\rfloor \quad \left\lfloor \frac{\vec{h}}{2} \right\rfloor \quad \vec{h} \right)$$

Note that:

$$G[\vec{h}]T_\ell = \left(\begin{array}{ccccc} \left\lfloor \frac{\vec{h}}{2^{\ell-1}} \right\rfloor & \dots & \left\lfloor \frac{\vec{h}}{2^{i}} \right\rfloor - 2\left\lfloor \frac{\vec{h}}{2^{i+1}} \right\rfloor & \dots & \vec{h} - 2\left\lfloor \frac{\vec{h}}{2} \right\rfloor \end{array} \right)$$

Which is just the binary representation of $\vec{h}$. Similarly, for a matrix $H = \left(\begin{array}{c|c|c} \vec{h}_1 & \dots & \vec{h}_t \end{array} \right)$, define:

$$G[H] = \left(\begin{array}{c|c|c} G[\vec{h}_1] & \dots & G[\vec{h}_t] \end{array} \right)$$

Then, for $\ell_i = \log \|\vec{h}_i\|_\infty$, we set

$$U = \begin{pmatrix} T_{\ell_1} & & \\ & \ddots & \\ & & T_{\ell_t} \end{pmatrix}$$

The corresponding $G[H] \times U$ is a zero-one matrix. Recall that for a given H, we would like to get $GP = H$, where P is also small. We thus set $G = G[H]$, and choose P to be a block-diagonal zero-one matrix, which selects the rightmost column of every block $G[\vec{h}_i]$. That is, for $\vec{p}_i = (0, \dots, 0, 1)^T$ of dimension i, set:

$$P = \begin{pmatrix} \vec{p}_{\ell_1} & & \\ & \ddots & \\ & & \vec{p}_{\ell_t} \end{pmatrix}$$

So that

$$G[H] \times P = \left(\begin{array}{c|c|c} G[\vec{h}_1] \times \vec{p}_{\ell_1} & \dots & G[\vec{h}_t] \times \vec{p}_{\ell_t} \end{array} \right) = \left(\begin{array}{c|c|c} \vec{h}_1 & \dots & \vec{h}_t \end{array} \right)$$

To satisfy $H = GP \in \Lambda_q^\perp(A_1)$, we choose H to be any basis of $\Lambda_q^\perp(A_1)$ (e.g. $H = \mathsf{HNF}(\Lambda_q^\perp(A_1))$. Now, set $A_2 = -A_1 \times G[H]$, and get:

$$\left(\begin{array}{c|c} A_1 & A_2 \end{array} \right) S = \left(\begin{array}{c|c} A_1 & A_2 \end{array} \right) \left(\begin{array}{c|c} G[H] \times U & 0 \\ \hline U & P \end{array} \right) = \left(\begin{array}{c|c} (A_1 G - A_1 G)U & -A_1 GP \end{array} \right) = 0 \pmod q$$

So what did we achieve so far? At this point, given $A_1 \in \mathbb{Z}^{n \times m_1}$, we can extend it with $A_2 \in \mathbb{Z}^{n \times m_2}$ and find a small $S \in \{-2, 0, 1\}^{m \times m}$, such that $\left(\begin{array}{c|c} A_1 & A_2 \end{array} \right) S = 0 \pmod q$. However, A_2 is completely determined by A_1, can we get back to $A_2 = -A_1 R$, for a random R, so that A_2 will be (close to) random given A_1?

Randomizing the matrix. Instead of setting $A_2 = -A_1 G$, let us set $A_2 = -A_1(G + R)$, where R is random. This already guarantees (by LOHL) that $\left(\begin{array}{c|c} A_1 & A_2 \end{array} \right)$ is close to random. Now, we adapt the rest of the construction accordingly. We require that

$$\left(\begin{array}{c|c} A_1 & A_2 \end{array} \right) \left(\begin{array}{c|c} (G+R)U & W \\ \hline U & P \end{array} \right) = 0 \pmod q$$

Which already zeros out the left part of the product. For the right part, we should zero out

$$A_1 W + A_2 P = A_1 W - A_1 (G + R) P$$

Choosing G and P as before, it holds that $A_1 GP = 0$, and hence to zero out the above, it suffices to set $W = RP$. It is left to check: (a) S is still small; (b) S is indeed a basis. The first check follows easily. Indeed, since R is a zero-one matrix and P simply selects a subset of its columns, then W is also a zero-one matrix. In addition, $(G + R)U = GU + RU$ is also small, since GU is small as before, and RU has entries of magnitude at most 3. We now show the second.

Claim 1. *S is a basis of $\Lambda_q^\perp(A)$ iff H is a basis of $\Lambda_q^\perp(A_1)$.*

Proof. Using linear-algebraic facts regarding the determinant of block matrices, we get for an invertible U:

$$\det S = \det \left(\begin{array}{c|c} V & W \\ \hline U & P \end{array} \right) = \det U \det \left(VU^{-1}P - W \right) = 1 \cdot \det((G+R)P - W) = \det GP = \det H$$

Now since both A_1 and A have full rank n, then $\det \Lambda_q^{\perp}(A_1) = \det \Lambda_q^{\perp}(A) = q^n$. Hence, S is a basis for $\Lambda_q^{\perp}(A)$ iff $\det S = q^n$ iff $\det H = q^n$ iff H is a basis for $\Lambda_q^{\perp}(A_1)$. □

Parameters. We started with $A_1 \in \mathbb{Z}^{n\times m_1}$, where $m_1 = \Omega(n \log q)$ (allowing use of LOHL). H has entries as large as q and so the number of columns in $G[H]$ is $m_2 \le m_1 \log q = O(n \log^2 q)$. Consequently, $m = m_1 + m_2 = O(n \log^2 q)$. The entries of S are all bounded by a constant and hence all vectors in S are of size $O(\sqrt{m})$.

Variants.

1. Instead of setting $GP = H \in \Lambda_q^{\perp}(A_1)$ in the above construction, set $GP = H - \Delta$ for some fixed Δ, and use $G[H-\Delta]$ rather than $G[H]$. Like the original construction, this construction can also be shown to satisfy our requirements. It turns out that for some choices of Δ (e.g. $\Delta = I$) result in improved parameters.

2. Alwen-Peikert also show a slightly different technique that achieves $m = O(n \log q)$. Their idea is to represent rows of H rather than columns, and use the fact that H has many small rows.

The Miccancio-Peikert Construction

Generate a random A with a trapdoor T that allows sampling random "short" vectors $\vec{x}$ such that $A\vec{x} = \vec{u} \pmod q$ for any given $\vec{u}$. This is done in two steps: (1) start from a special lattice $G \in \mathbb{Z}^{n\times m_1}$, for which the above sampling is possible; (2) Use the trapdoor to translate the random A to the special G.

For a matrix $B \in \mathbb{Z}^{n\times m}$, denote $f_B(\vec{x}) = A\vec{x} \pmod q$. Our goal is to generate A with a trapdoor T that allows sampling short pre-images of a given $\vec{u}$ under f_A.

Step 1: In homework. Yields $G \in \mathbb{Z}_q^{m_2}$, where $m_2 = n \lceil \log q \rceil$.

Step 2: Choose $A_1 \in_R \mathbb{Z}_q^{n\times m_1}$, where $m_1 = \lceil 3n \log q \rceil$. Set $A_2 = -A_1 R + G \pmod q$ for $R \in_R \{0,1\}^{m_1 \times m_2}$. Output the matrix and trapdoor

$$A = \left(\begin{array}{c|c} A_1 & A_2 \end{array} \right) \qquad T = \begin{pmatrix} I & R \\ 0 & I \end{pmatrix}$$

Sampling: given $\vec{u} \in \mathbb{Z}_q^n$, do the following:

1. Sample a short $\vec{z}_1 \in \mathbb{Z}^{m_1}$ (e.g. from a sphere or Gaussian).

2. Set $\vec{v} = \vec{u} - A_1 \vec{z}_1 \pmod q$.

3. Sample a short pre-image $\vec{z}_2$ of $\vec{u}$ under f_G.

4. Output $\vec{w} = \begin{pmatrix} \vec{w}_1 \\ \vec{w}_2 \end{pmatrix} = T \begin{pmatrix} \vec{z}_1 \\ \vec{z}_2 \end{pmatrix} = \begin{pmatrix} \vec{z}_1 + R\vec{z}_2 \\ \vec{z}_2 \end{pmatrix}$

$\vec{z}_1, \vec{z}_2$ are short by construction, and so is R; hence, $\vec{w}$ is short. In addition,

$$A\vec{w} = \left(\begin{array}{c|c} A_1 & G - A_1 R \end{array} \right) \begin{pmatrix} I & R \\ 0 & I \end{pmatrix} \begin{pmatrix} \vec{z}_1 \\ \vec{z}_2 \end{pmatrix} = \left(\begin{array}{c|c} A_1 & G \end{array} \right) \begin{pmatrix} \vec{z}_1 \\ \vec{z}_2 \end{pmatrix} =$$

$$A_1\vec{z}_1 + G\vec{z}_2 = A_1\vec{z}_1 + \vec{v} = A_1\vec{z}_1 + (\vec{u} - A_1\vec{z}_1) = \vec{u} \pmod q$$

Remark: If $\vec{z}_1, \vec{z}_2$ are chosen from a spherical distribution, $\vec{w}$ is chosen from a "skewed" distribution, due to the effect of T (which can be fixed with some extra effort).

References

[Ajt99] Miklós Ajtai. Generating hard instances of the short basis problem. In *ICALP'99*, volume 1644 of *Lecture Notes in Computer Science*, pages 1–9. Springer, 1999.

[AP11] Joël Alwen and Chris Peikert. Generating shorter bases for hard random lattices. *Theory Comput. Syst.*, 48(3):535–553, 2011.

[MP11] Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In *manuscript*, 2011.