Lattices and Homomorphic Encryption, Spring 2013 Instructors: Shai Halevi, Tal Malkin

Problem Set #6

April 23, 2013 Due May 7, 2013

1 LWE-Based Homomorphic Encryption with Packed Ciphertexts

In this question we try to make the LWE-Based Homomorphic Encryption that we saw in class more efficient, by "packing" many plaintext bits in a single ciphertext. Recall that in that cryptosystem, plaintexts are bits $b \in \{0,1\}$, and ciphertexts and secret keys are n-vectors, $\vec{c}, \vec{s} \in \mathbb{Z}^n$, where $\|\vec{c}\|_\infty \leq q/2$ $\|\vec{s}\|_\infty \ll q$. A ciphertext $\vec{c}$ is a valid encryption of $b \in \{0,1\}$ relative to key $\vec{s}$ if their inner product (over the integers) satisfies

$$\langle \vec{s}, \vec{c} \rangle = q \cdot k + \lceil q/2 \rceil \cdot b + e$$

where $|k|, |e| \ll q$.

Recall also that we denote $n' = n - 1$, and the secret-key $\vec{s}$ has the form $\vec{s} = (\vec{s}'|1)$ for some secret vector $\vec{s} \in \mathbb{Z}^{n'}$. The public key consists of a random matrix $A' \in \mathbb{Z}_q^{n' \times m}$ and another m-vector $\vec{a} = -\vec{s}'A' + \vec{e}' \bmod q$ (with $\|\vec{e}'\| \ll q$). Adding $\vec{a}$ to A' as the n'th row we get a matrix A such that $\vec{s}A = \vec{e} \pmod q$.

To reduce the plaintext-to-ciphertext expansion ratio, we can encrypt any number t of plaintext bits in a single ciphertext, by increasing the dimension of the ciphertext from $n'+1$ to $n'+t$. The secret key will now be a matrix rather than a single vector, and the decryption formula becomes

$$S\vec{c} = q \cdot \vec{k} + \lceil q/2 \rceil \cdot \vec{b} + \vec{e} \tag{1}$$

where $\vec{b} \in \{0,1\}^t$ is the plaintext vector and $\|\vec{k}\|, \|\vec{e}\| \ll q$.

Specifically, we choose t vectors from the error distribution, $\vec{s}_i' \leftarrow \mathcal{D}_{\mathbb{Z}^{n'},\sigma}$, and set them as the rows of a matrix $S' \in \mathbb{Z}^{t \times n'}$. As before we choose a random matrix $A' \in \mathbb{Z}_q^{n' \times m}$, and we set $\vec{a}_i = -\vec{s}_i' A' + \vec{e}_i$ (where the error vectors $\vec{e}_i$ are chosen from the error distribution). Adding the $\vec{a}_i$'s to A' as rows $n'+1, \ldots, n'+t$ we get the public key matrix $A \in \mathbb{Z}_q^{t \times (n'+t)}$. Also adding to S' the columns of the $t \times t$ identity matrix, we get $S = (S'|I) \in \mathbb{Z}^{t \times (n'+t)}$, and we have $S \times A = E \pmod q$ where E is the matrix that has the $\vec{e}_i$'s for rows.

1. Describe the encryption procedure, and prove that using the decryption formula from Equation 1 indeed recovers the plaintext.

2. Prove security for this encryption scheme, assuming the hardness of D-LWE.

   Hint. You can reduce directly to D-LWE by arguing that A is pseudorandom and the attacker would have no information on the plaintext if it was random (due to the leftover hash lemma). Alternatively you can use the fact that we proved in class the security of the variant with $t = 1$, and reduce the security for general t to the security of $t = 1$ using a hybrid argument.

3. Prove that the scheme is additively homomorphic.

4. Prove that the scheme as-is supports a constant number of homomorphic entry-wise multiplications, via tensoring. That is, given two ciphertexts $\vec{c}_1, \vec{c}_2$, encrypting under S the vectors $\vec{b}_1, \vec{b}_2$, respectively, compute a ciphertext $\vec{c}^*$ that encrypts under some S^* the vector $\vec{b}^*$ such that $\vec{b}^*[i] = \vec{b}_1[i] \cdot \vec{b}_2[i]$.

5. (bonus) Show how to add a key-switching gadget to the public key to reduce the dimension of the tensored ciphertext vectors.

   Hint. Recall that for the case of $t = 1$, the key-switching gadget included many vectors that roughly "encrypt" the entries of $\vec{s}^*$ under $\vec{s}$. Extend it to a general t by "encrypting" the columns of S^* under S.