6.876J Advanced Topics in Cryptography: Lattices	Oct 14, 2015
Lecture 10	
Lecturer: Vinod Vaikuntanathan	*Scribe: Lauren de Meyer, Prashant Vasudevan*

Last class we showed the **NP**-hardness of the Closest Vector Problem (CVP). Today, we discuss the **NP**-hardness of the Shortest Vector Problem (SVP) and its gap version, GapSVP.

1 History

The essential parts of the history of hardness proofs for SVP are presented in Table 1. By SVP_γ and CVP_γ, we denote GapSVP and GapCVP respectively with gap γ. **NP**-hard* means that the reduction is randomised - an algorithm for this problem would give a randomised algorithm for any problem in **NP**. Quasi-**NP**-hard means the reduction runs in super-polynomial (though still sub-exponential) time.

Problem	Norm	Hardness	Reference
CVP_1	ℓ_p	NP-hard	[vEB81]
SVP_1	ℓ_∞	NP-hard	[vEB81]
SVP_1	ℓ_2	NP-hard*	[Ajt98]
$\mathsf{SVP}_{\sqrt{2}}$	ℓ_2	NP-hard*	[Mic98]
SVP_c	ℓ_p	NP-hard*	[Kho04]
$\mathsf{SVP}_{2^{(\log n)^{1/2-\epsilon}}}$	ℓ_p	quasi-NP-hard	[Kho04]
$\mathsf{SVP}_{2^{(\log n)^{1-\epsilon}}}$	ℓ_p	quasi-NP-hard	[HR07]

Table 1: History of **NP**-hardness proofs for lattice problems.

The following questions immediately arise from the current state of knowledge of **NP**-hardness reductions as presented in Table 1, and are still open.

1. Can the reductions to SVP with any gap at all be derandomised? See [Mic12] for notes on analogies to similar problems from the domain of linear codes.
2. It was shown in [AR05] that $\mathsf{SVP}_{n^{1/2}}$ is in **coNP**. This makes it unlikely that $\mathsf{SVP}_{n^{1/2}}$ is **NP**-hard. But what about $\mathsf{SVP}_{n^{1/2-\epsilon}}$ for some ϵ?
3. Can we get polynomial time reductions for the cases where we now have sub-exponential time reductions (from [Kho04] and [HR07])? If so, then things here would mirror the state of affairs for the same gaps for CVP.

2 NP-hardness of SVP in ℓ_2-norm

We present here the reduction from [Kho04] SVP_c for constant c. While [Kho04] works for SVP in the ℓ_p for any $p > 1$, we shall concern ourselves only with the ℓ_2 norm. It is to be mentioned that the reduction due to [HR07] mentioned in Table 1 starts with this reduction and uses tensor products to amplify the gap.

The reduction proceeds in the following three steps:

1. $\mathsf{ESC} \leq \mathsf{CVP}^*$

2. $\mathsf{CVP}^* \leq \mathsf{SVP}^*$
3. $\mathsf{SVP}^* \leq \mathsf{SVP}$

The problems ESC (Exact Set Cover), CVP^*, and $\mathsf{SVP}^*$are defined when discussing the respective steps below.

Step 1: $\mathsf{ESC} \leq \mathsf{CVP}^*$

Exact Set Cover. For each n', n'', the promise problem $\mathsf{ESC}(\gamma, d)$ is defined over sets of n'' subsets $S_1, \ldots, S_{n''} \subseteq [n']$ (along with n') by the following characterisation of YES and NO instances:

- YES: There is a collection of γd subsets among $S_1, \ldots, S_{n''}$ that partition $[n']$.
- NO: No collection of $\leq d$ subsets among $S_1, \ldots, S_{n''}$ covers $[n']$.

$\mathbf{CVP}^*$. The promise problem $\mathsf{CVP}^*(\gamma, d)$ is defined over pairs of full rank matrices and vectors (B, t) by the following characterisation of YES and NO instances:

- YES: $\exists y \in \Lambda(B)$ such that $(y - t)$ is a 0-1 vector with at most γd ones
- NO: $\forall y \in \Lambda(B)$ and $\beta \in \mathbb{Z} \backslash \{0\}$, $(y - \beta t)$ has at least d non-zero coefficients.

Reduction:

For any d and γ, we show how to reduce $\mathsf{ESC}(\gamma, d)$ to $\mathsf{CVP}^*(\gamma, d)$. Given an instance $(n', S_1, \ldots, S_{n''})$ of $\mathsf{ESC}(\gamma, d)$, let χ_{S_i} be the characteristic vector of S_i (of length n'). Then the CVP^*instance reduced to is (B_{CVP}, t), where B_{CVP} and t are constructed as follows:

1. Pick matrix S as:
$$S \leftarrow \begin{bmatrix} \vdots & \vdots & \vdots \\ \chi_{S_1} & \ldots & \chi_{S_{n''}} \\ \vdots & \vdots & \vdots \end{bmatrix} \in \{0,1\}^{n' \times n''}$$
2. Let L_{CVP} be the lattice defined as $L_{CVP} = \{z \in \mathbb{Z}^{n''} \mid Sz = 0\}$. Pick B_{CVP} to be a basis of L_{CVP}.
3. Pick t to be any vector in $\mathbb{Z}^{n''}$ such that $St = -\overrightarrow{1}$.

In other words, B_{CVP} is a basis of the dual of the lattice formed by the characteristic vectors of the sets $S_1, \ldots, S_{n''}$, and t is a vector in a certain coset of this lattice. The following argue that the reduction is correct:

- If $(n', S_1, \ldots, S_{n''})$ is a YES instance of $\mathsf{ESC}(\gamma, d)$, this implies that there is a 0-1 vector z with at most γd 1's such that $Sz = \overrightarrow{1}$. Then the vector $y = z + t$ is in $\Lambda(B_{CVP})$ because $Sy = S(z + t) = \overrightarrow{0}$, and $(y - t) = z$ is a 0-1 vector with at most γd 1's. Thus, (B_{CVP}, t) is a YES instance of $\mathsf{CVP}^*(\gamma, d)$.
- If $(n', S_1, \ldots, S_{n''})$ is a NO instance of $\mathsf{ESC}(\gamma, d)$, then no collection of at most d subsets even covers $[n']$. For any $y \in \Lambda(B_{CVP})$ and $\beta \in \mathbb{Z} \backslash \{0\}$, note that $S(y - \beta t) = -\beta St = \beta \overrightarrow{1}$. So $(y - \beta t)$ cannot have less than d non-zero co-efficients as the set of subsets corresponding to the non-zero co-efficients of $(y - \beta t)$ constitutes a cover of $[n']$. Hence, (B_{CVP}, t) is a NO instance of $\mathsf{CVP}^*(\gamma, d)$.

Step 2: $\mathsf{CVP}^* \leq \mathsf{SVP}^*$

SVP*. The promise problem $\mathsf{SVP}^*(\zeta, d)$ is defined over a full rank matrix B by the following characterisation of YES and NO instances:

- YES: There are "many" vectors $z \in \Lambda(B)$ with $\|z\| \leq \sqrt{\zeta d}$
- NO: There are only "a few" vectors $z \in \Lambda(B)$ with $\|z\| < d$.

We leave "many" and "a few" undefined for now. We will reduce $\mathsf{CVP}^*(\gamma, d)$ to $\mathsf{SVP}^*(4\gamma + \frac{r}{d}, d)$ (for r to be specified later) by creating a lattice that has a guaranteed number of short vectors ($\|\cdot\| \leq \sqrt{4\gamma d + r}$) when the CVP^* instance is a YES and only a limited number of vectors with length $< d$ when the CVP^* instance is a NO.

A First Attempt

Consider a CVP^*instance (B_{CVP}, t) with $y = B_{CVP}w$ the solution. First, construct the lattice basis $B = \begin{bmatrix} B_{CVP} & t \end{bmatrix}$.

- If (B_{CVP}, t) is a YES instance of $\mathsf{CVP}^*(\gamma, d)$ then t is close to the lattice L_{CVP}. This implies the existence of a vector $w \in \{0,1\}^{n''}$ such that $y = B_{CVP}w$ is the solution of $\mathsf{CVP}^*$and thus $\begin{bmatrix} B_{CVP} & t \end{bmatrix} \begin{bmatrix} w \\ -1 \end{bmatrix} = y - t$ is a 0-1 vector with at most γd ones $\Leftrightarrow \|\cdot\|_2 \leq \sqrt{\gamma d}$.
- If (B_{CVP}, t) is a NO instance of $\mathsf{CVP}^*(\gamma, d)$, then for any lattice vector $y = B_{CVP}w$ and for any $\beta \in \mathbb{Z}\backslash\{0\}$: $\begin{bmatrix} B_{CVP} & t \end{bmatrix} \begin{bmatrix} w \\ \beta \end{bmatrix} = y + \beta t$ has at least d non-zero coefficients $\Leftrightarrow \|\cdot\|_2 \geq \sqrt{d}$.

What about the case when $\beta = 0$? For this, consider the SVP lattice from $\begin{bmatrix} B_{CVP} & t \\ L & s \end{bmatrix}$ with L a gadget lattice and s some point that must satisfy the following properties:

1. s is close to all Lw vectors with $w \in \{0,1\}^{n''}$ such that the YES-instances

$$\begin{bmatrix} B_{CVP} & t \\ L & s \end{bmatrix} \begin{bmatrix} w \\ -1 \end{bmatrix} = \begin{bmatrix} B_{CVP}w - t \\ Lw - s \end{bmatrix}$$

   are not ruined.

2. For NO-instances $\begin{bmatrix} B_{CVP} & t \\ L & s \end{bmatrix} \begin{bmatrix} w \\ \beta \end{bmatrix} = \begin{bmatrix} B_{CVP}w + \beta t \\ Lw + \beta s \end{bmatrix}$, we have either:
   - $\beta \neq 0 \Rightarrow$ the vector $B_{CVP}w + \beta t$ already has at least d non-zero coefficients by CVP^*.
   - $\beta = 0 \Rightarrow$ we must make sure Lw is not short

Finding L and s

Consider a binary $[n, k, d]_2$ code that transforms k-bit messages to n-bit codewords such that the minimum distance between any two codewords is d. With G the generator matrix of the code, let L be the basis of the lattice generated by $\begin{bmatrix} G & 2I \end{bmatrix}$ (the basis vectors of G mod 2).

Lemma 1. *Every vector in $\Lambda(L)$ has either*

- *only even coordinates (= vectors corresponding to the zero codeword)*
- *$\geq d$ non-zero coordinates (=vectors corresponding to non-zero codewords)*

Lemma 2. *For any $r \in \mathbb{N}$, there is a point $s \in \{0,1\}^n$ such that the number of lattice vectors in $\Lambda(L)$ at distance at most r from s is at least:*

$$\frac{2^k}{2^n}\binom{n}{r} = \frac{1}{2^{n-k}}\binom{n}{r}$$

Proof. Consider the collection of 2^k codewords (that are also vectors in $\Lambda(L)$) and the space $\{0,1\}^n$ of 2^n vectors. Connect each codeword to the vectors with hamming distance r (which are obtained by flipping r bits of the codeword). In each codeword one can choose $\binom{n}{r}$ different combinations of bits to flip so each codeword has $\binom{n}{r}$ edges. This implies that the total number of edges is $\binom{n}{r} \cdot 2^k$. So there is at least one point $s \in \{0,1\}^n$ that has at least $\binom{n}{r}\frac{2^k}{2^n}$ incident edges, which implies there are at least so many points in $\Lambda(L)$ within distance r from it. $\square$

The Reduction

Now we can finish our reduction from $\mathsf{CVP}^*(\gamma, d)$ to SVP^* by dealing with the case of $\beta = 0$. Consider the following lattice basis:

$$B_{SVP} = \begin{bmatrix} 2B_{CVP} & 0 & 2t \\ 0 & L & s \end{bmatrix}$$

- If (B_{CVP}, t) is a YES instance of $\mathsf{CVP}^*(\gamma, d)$, then there are vectors y, w such that $y - t = B_{CVP}w - t$ is a 0-1 vector with at most γd ones. By Lemma 2, there exists a vector s and at least $\frac{1}{2^{n-k}}\binom{n}{r}$ vectors w' such that $Lw' - s$ has at most r ones. This means that there exist at least $\frac{1}{2^{n-k}}\binom{n}{r}$ lattice vectors $z = B_{SVP}\begin{bmatrix} w \\ w' \\ -1 \end{bmatrix} = \begin{bmatrix} 2(B_{CVP}w - t) \\ Lw' - s \end{bmatrix}$ with $\|z\|_2 \leq \sqrt{4\gamma d + r}$ (We call these z's **good vectors**).

- If (B_{CVP}, t) is a NO instance of $\mathsf{CVP}^*(\gamma, d)$, then consider the vector $z = B_{SVP}\begin{bmatrix} w \\ w' \\ \beta \end{bmatrix}$ for any w, w' and β. We have the following cases:
  - $\beta \neq 0 \Rightarrow$. In this case, the fact that this is a NO instance of CVP^* guarantees that $\|z\|_2 \geq \sqrt{d}$.
  - $\beta = 0$. In this case, the lattice vectors are of the form $z = \begin{bmatrix} 2B_{CVP}w \\ Lw' \end{bmatrix}$.
    1. If $\exists$ an odd coordinate in z, then by Lemma 1, there are $\geq d$ odd coordinates $\Leftrightarrow$ $\|z\|_2 \geq d$.
    2. Else, there are only even coordinates:
       * If there are at least $\frac{d}{4}$ non-zero coordinates in z, then $\|z\|_2 \geq d$.
       * Else, z has less than $\frac{d}{4}$ non-zero coordinates:
         · If there is a coordinate with absolute value at least d, then $\|z\|_2 \geq d$
         · Else, we call z an **annoying vector**. Such a vector has only a few ($< d/4$) non-zero coordinates and each coordinate have absolute value at most d. By straightforward counting, we can see that the number of **annoying vectors** is at most:

$$\binom{n + n'' - n'}{d/4}(2d+1)^{d/4}$$

If one chooses $r = (\frac{3}{4} + \epsilon)d$ for some small constant ϵ and uses a BCH code ($[n, n - \frac{d}{2}\log n, d]_2$) in the above reduction, then one can make sure that the number of annoying vectors in any instance that a NO instance of CVP^* reduces to is much smaller than the number of good vectors in any instance that a YES instance reduces to.

Note that there was a non-uniform step involved in the reduction, which was knowing an s such that there are a lot of codewords close to it as predicted by Lemma 2. This non-uniformity can be traded for randomness, as it can be shown that a random 0-1 vector, in fact, has very similar properties.

Step 3: $\mathsf{SVP}^* \leq \mathsf{SVP}_c$

We reduce SVP^* to SVP with (an arbitrarily large) constant gap by transforming the lattice with basis B_{SVP} to a new lattice $L_{w,q} = \{v \in \Lambda(B_{SVP}) \text{ s.t. } \langle v, w\rangle = 0 \bmod q\}$. For any set of vectors in $\Lambda(B_{SVP})$, with high probability (over the choice of w), 1 in q vectors from this set is present in $L_{w,q}$. So by choosing q appropriately and choosing w at random, we can ensure that with high probability, none of the annoying vectors are retained when reducing from a NO istance of SVP^*, and least 1 good vector is retained when reducing from a YES instance.

References

[Ajt98] Miklós Ajtai. The shortest vector problem in L_2 is NP-hard for randomized reductions (extended abstract). In *Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing, Dallas, Texas, USA, May 23-26, 1998*, pages 10–19, 1998.

[AR05] Dorit Aharonov and Oded Regev. Lattice problems in NP cap conp. *J. ACM*, 52(5):749–765, 2005.

[HR07] Ishay Haviv and Oded Regev. Tensor-based hardness of the shortest vector problem to within almost polynomial factors. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing, San Diego, California, USA, June 11-13, 2007*, pages 469–477, 2007.

[Kho04] Subhash Khot. Hardness of approximating the shortest vector problem in lattices. In *45th Symposium on Foundations of Computer Science (FOCS 2004), 17-19 October 2004, Rome, Italy, Proceedings*, pages 126–135, 2004.

[Mic98] Daniele Micciancio. The shortest vector in a lattice is hard to approximate to within some constant. In *39th Annual Symposium on Foundations of Computer Science, FOCS '98, November 8-11, 1998, Palo Alto, California, USA*, pages 92–98, 1998.

[Mic12] Daniele Micciancio. Inapproximability of the shortest vector problem: Toward a deterministic reduction. *Theory of Computing*, 8(1):487–512, 2012.

[vEB81] Peter van Emde Boas. Another np-complete partition problem and the complexity of computing short vectors in a lattice. Technical Report 81-04, Mathematische Instituut, University of Amsterdam, 1981.