6.841/18.405J: Advanced Complexity Theory	September 5, 2001
Lecture 21	
Lecturer: Madhu Sudan	*Scribe: Ashish Mishra*

Today's lecture:

- A DNP-Complete Problem
- Lattice Problems and Worst-Case vs. Average-Case complexity: connection to NP

1 Recap from last lecture

We present a high-level view of the notion of average-case complexity. Some progress has been made in the last 10-15 years on analyzing these problems, including connecting problems of worst-case and average-case complexity, and developing the first cryptosystem relying on worst-case, rather than average-case complexity.

In the last lecture we talked about DNP (Distributed NP).

Definition 1 *DNP is the class of problems specified by (R, D) where R is a poly time computable binary relation and $D : \{0,1\}^n \to \{0,1\}^n$ is a poly time computable function.*

Effectively, D specifies the distribution induced by applying the function to a uniformly chosen random input, $\{D(z)\}_{z \in_U \{0,1\}^n}$. In practice we won't concern ourselves with the exact length of the random input since random lengths can be mapped to each other to a good extent.

Definition 2 *An algorithm A is δ-good for (R, D) if A solves $(1-\delta)$ fraction of the instances of R drawn according to D, never giving a wrong answer.*

Definition 3 *$(R, D) \in$ Avg-P iff there is a $B(x, \delta)$ such that $A_\delta(.) = B(., \delta)$ is δ-good for (R, D) and $B(x, \delta)$ runs in poly$(|x|, 1/\delta)$ time.*

This definition is drawn from Impagliazzo's survey.

The idea of reductions is straightforward: we pick a random z, and give instance $D(z)$. The reduction specifies a particular distribution. To relate different distributions, we introduce the notion of "domination".

Definition 4 *Distribution D_1 α-dominates distribution D_2 if $\forall x$,*

$$\Pr_{D_1}[x] \geq \frac{\Pr_{D_2}[x]}{\alpha}$$

Domination is a very one-sided relationship and it is easy to construct distributions D_1, D_2 as in Figure 1 where D_1 α-dominates D_2 — but D_2 doesn't α'-dominate D_1 for any finite α'.

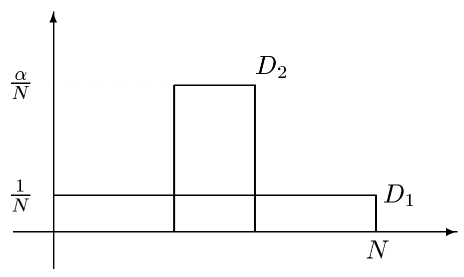

Figure 1: Dominating distributions

Theorem 5 *If algorithm A is δ-good for (R, D_1) and D_1 α-dominates D_2, then A is $(\alpha\delta)$-good for (R, D_2).*

The Impagliazzo-Levin Lemma states that every (R, D) problem reduces to some (R', U) where U is an essentially uniform distribution.

Consider R' which is the composition $R \circ D$. To implement this, we need to have a uniformly randomly chosen pre-image of x under D. Computing the inverse of D may be hard, so instead we specify z implicitly: by giving an index w of z within the pre-images of x.

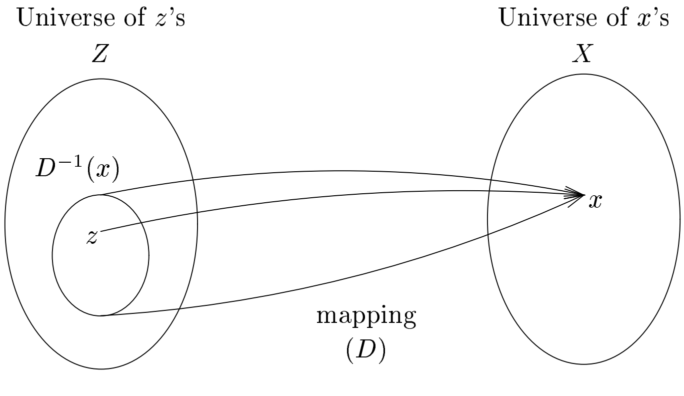

Figure 2: Picking a Distribution

We are picking $x \in \{0,1\}^n, w \in \{0,1\}^k$, where x has roughly 2^k pre-images under D. x and w specify $z \in \{0,1\}^n$, and we would like the distribution of z to be nearly uniform. We can do this by hashing the $(n+k)$ length string (x, w) down to an n-length string. The hash is drawn uniformly from a family of pairwise independent hash functions.

Formally R' is expressed as a tuple (u, k, h_1, h_2) where $u \in \{0,1\}^n$ is the hash of (x, w), $k \in \{0, 1...n\}$ is the log of the number of pre-images of x, and $h_1 : \{0,1\}^n \to \{0,1\}^k$, $h_2 : \{0,1\}^{n+k} \to \{0,1\}^n$ are hash functions. $R'((u, k, h_1, h_2), (z, y))$ iff:

- y is a witness to z under R, i.e. $R(D(z), y)$.
- $u = h_2(D(z), h_1(z))$.

Our distribution on U is to pick u, k uniformly from $\{0,1\}^n$ and $\{0, 1...n\}$ respectively, and h_1, h_2 uniformly from the families of hash functions on that many bits. The tricky part is guessing the size of the pre-image correctly. But with a random guess we have a $\frac{1}{n+1}$ chance of getting the right value, within a factor of 2.

2 Proof of the Impagliazzo-Levin Lemma

2.1 Reduction

Given x,

1. Uniformly pick $k \in_U \{0, 1, ...n\}$. (so we have $\frac{1}{n+1}$ chance of picking the right value)
2. Uniformly pick $w \in_U \{0,1\}^k$.
   We don't yet know z or a distribution for z, but we can stipulate that $w = h_1(z)$.
3. Uniformly pick hash functions h_1, h_2.
4. Output: $(u = h_2(x, w), k, h_1, h_2)$.

If we have accurately estimated the number of pre-images 2^k above, then with high probability there will exist unique z such that $D(z) = x$ and $h_1(z) = w$.

2.2 Analysis

We can construct a corresponding distribution D_2 on (u, k, h_1, h_2) as follows:

1. Uniformly pick $z \in_U \{0,1\}^n$.
2. Set $k = \log_2 |D^{-1}(D(z))|$.

   We don't know how to compute this, we are just defining a distribution for the purpose of analysis.
3. Uniformly pick hash functions h_1, h_2.
4. Output: $(u = h_2(D(z), h_1(z)), k, h_1, h_2)$.

We make the following claims on D_2:

- The distribution U $O(n)$-dominates D_2.
- (R', D_2) is at least as hard as (R, D).

This completes the proof that given (R, D), there exists a problem (R', U) which is equally hard.

However, the above does not provide us with a single hard problem independent of R and D. To do this, we create a universal relation:

$$R_U \left\langle \begin{array}{c} (R,x) \\ y \end{array} \right.$$

with $R_U((R,x), y)$ holding iff $R(x,y)$ holds. We need not worry what distribution to apply to R_U — as long as x is chosen with finite probability, any distribution suffices.

This means that every interesting problem (e.g. factoring, SAT) has a uniform distribution describing it.

3 Lattice problems

DNP gives us a theory of average-case problems. Effectively, it relates average-case problems to each other. But it does *not* relate average-case to worst-case problems. What might be desirable is a connection analogous to that between approximation problems and exact calculation problems. Currently we don't know of any such result relating worst-case and average-case hardness in NP.

In 1996, Ajtai showed the existence of a lattice problem R' that we don't know how to solve in RP; and an instance (R, D) of Avg-P such that R' is reducible to (R, D). Essentially this gives a reduction from a "hard" problem in worst-case complexity to a related problem in average-case complexity. If R' was known to be NP-complete, this would be a dream theorem in our current context. As it is, it represents a major breakthrough in classifying average-case complexity.

3.1 Definitions

Lattices consist of discrete points in $\mathbb{R}^n$, in some regular symmetric form. Figure 3 illustrates an example with only 2 dimensions for pictorial effect.

It is important to distinguish this geometric notion of lattices from the algebraic entities involving partially ordered sets, as used in other contexts. Mathematically

Definition 6 *A lattice L is a discrete additive subset of $\mathbb{R}^n$.*

Discrete: There is some $d > 0$ such that for any $x \in L$, $\text{Ball}(x, d) \cap L = \{x\}$

Additive: For any $x, y \in L$, it follows that $x + y, x - y \in L$

If L is nonempty then it clearly must contain the origin, since from additivity $x - x \in L$. There are two ways to specify a lattice, known as the *primal* and the *dual* representations.

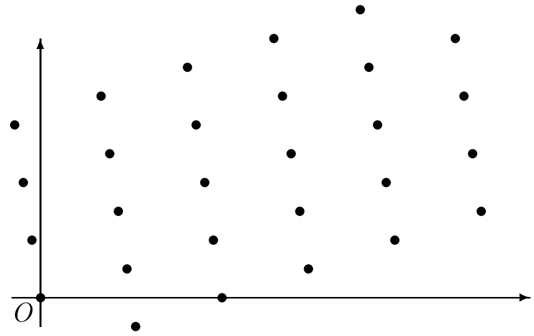

Figure 3: 2-dimensional lattice

3.1.1 Primal Representation

This gives a basis for the lattice: $b_1, b_2, ...b_m \in \mathbb{R}^n$, with the b_i's linearly independent ($\Rightarrow m \leq n$). Then

$$L(b_1, ...b_m) = \left\{ \sum_{i=1}^{m} z_i b_i \mid z_i \in \mathbb{Z}\ \forall i \right\}$$

We often think about lattices on $\mathcal{Q}^n$, $\mathbb{Z}^n$ rather than $\mathbb{R}^n$.

3.1.2 Dual Representation

While the primal representation gives a constructive description of the lattice, this representation instead specifies a set of constraints on it. Mathematically we are given $b_1^*, b_2^*...b_m^*$ with $m \geq n$. Then

$$L(b_1^*, ...b_m^*) = \left\{ v \mid \langle v, b_j^* \rangle \in \mathbb{Z}\ \forall j \right\}$$

where $\langle v, b \rangle$ denotes the inner product of v and b.

If we have precisely n vectors, the dual basis vectors are simply given by the inverse of the matrix composed of basis vectors. In other cases, it is not easy to see what the dual basis vectors intuitively represent, but it is still algorithmically easy to go from one representation to another.

3.2 Problems relating to lattices

Easy Problems :

1. Computing intersection of two lattices (itself a lattice).
2. Computing bases of a lattice.

Hard Problems :

1. **Short Vector Problem (SVP):** Given a basis, does there exist a non-zero vector of length $\leq d$ in the lattice?

   Ajtai showed this problem to be NP-complete under randomized reductions.
2. Given a basis $b_1, b_2...b_m$, can you find a short basis of length $\leq d$ for the same lattice?
3. Given $b_1, b_2...b_m$, and a target $t \in \mathbb{R}^n$, what is the nearest vector to t in L? (NP-hard)

All these hard problems are optimization problems, so we can consider approximation algorithms to solve them. Lenstra, Lenstra and Lovasz gave a 2^n-approximation algorithm to SVP on an n-dimensional lattice. This was subsequently improved, but not significantly. We now have $2^{o(n)}$-approximations, but not $2^{\sqrt{n}}$.

Though this approximation might seem too weak to be any use, it actually has significant applications. LLL showed how to use it to factor integer polynomials in poly time. Other uses of lattice problems are in cryptanalysis, integer programming, diophantine systems, and even in building cryptosystems [Ajtai-Dvork]. This shows that lattice problems are of immense interest.

3.3 Ajtai's theorem

Ajtai's theorem involves polynomial approximations to the short vector/short basis problem. Ajtai compared the following two problems, and shows that an Avg-P solution to the average-case problem implies an RP solution to the worst-case problem.

3.3.1 Worst-case problem

Given $L(b_1, b_2 ... b_m) \subset \mathbb{R}^n$, find an n^{c_1}-approximate small basis for L.

Currently, there is no knowledge about the hardness of this problem.

3.3.2 Average-case problem

Given $L^*(b_1^*, b_2^* ... b_m^*) \subset \mathbb{R}^n$, find an n^{c_2}-approximate short vector in L^*.

The distribution D is as follows:

- Fix $q = n^{c_3}$
- Fix $m = \Theta(n \log q)$
- Randomly choose $(b_1^* ... b_m^*) \in_R \left\{0, \frac{1}{q}, \frac{2}{q}, ... \frac{q-1}{q}, 1\right\}^n$

L^* clearly is an infinite lattice, since $q\mathbb{Z}^n \subset L^*$.