Notes for Lecture 20

Last time, we went over the Goldreich-Levin sublinear-time list-decoding of the Hadamard code. Then for $p : \{0,1\}^k \to \{0,1\}^k$ a one-way permutation, we considered the extended permutation given by $x.r \mapsto p(x).r$, $\{0,1\}^{2k} \to \{0,1\}^{2k}$. Hadamard list-decoding implied $B(x,r) = x \cdot r \mod 2$ to be a hard-core predicate for the extended permutation.

In this lecture, we will prove a large class of predicates to be hard-core for the specific conjectured one-way permutations RSA and EXP, following Akawa, Goldwasser and Safia [AGS]. The construction will not require extending the permutations.

20.1 RSA and EXP

Definition 1 *Given $N = pq$, p and q prime, choose e so $gcd(e, \varphi(N)) = 1$. Then the* RSA *permutation $\mathbf{Z}_N \to \mathbf{Z}_N$ is*

$$\mathrm{RSA}_{N,e}(x) = x^e \mod N \ . \tag{1}$$

Definition 2 *Given p prime, g a generator for $\mathbf{Z}_p^*$, the* EXP *isomorphism $\mathbf{Z}_{p-1} \to \mathbf{Z}_p^*$ is*

$$\mathrm{EXP}_{p,g}(x) = g^x \mod p \ . \tag{2}$$

20.2 Accessible, list-decodable code implies hard-core predicate

Consider $p : \mathbf{Z}_N \to \mathbf{Z}_n$ a permutation, $B : \mathbf{Z}_N \to \{0,1\}$ a predicate, $C : \mathbf{Z}_N \to \{0,1\}^n$ a code.

Definition 3 *C is* accessible *with respect to p, B if there is an algorithm A such that*

- *For $x \in \mathbf{Z}_N$, $j \in \{1, \ldots, n\}$,*

$$A(p(x), j) = p(y) \quad s.t. \quad B(y) = C(x)_j \ . \tag{3}$$

- *Over the choice of random x and j, $A(p(x), j)$ is distributed close to uniform in $\mathbf{Z}_N$.*

Essentially, bits of the codeword are located in the table for B. So an algorithm for computing $B(y)$ from $p(y)$ can be used to determine bits of the encoding of $C(x)$.

Lemma 1 *Assume p is a one-way permutation, C is accessible (w.r.t. p,B), and C is list-decodable (from $\frac{1}{2} + \epsilon$ agreement in time* $\mathrm{poly}(\frac{1}{\epsilon}, \log N)$*). Then B is a hard-core predicate for p.*

Indeed, assume otherwise, that on $\frac{1}{2} + \epsilon$ fraction of the y, we can determine $B(y)$ from $p(y)$. Over x and j, $A(p(x), j)$ is (nearly) uniform, so there is some $\frac{\epsilon}{2}$ fraction of the x's for which $\frac{1}{2} + \frac{\epsilon}{2}$ of the codeword indices j are good. The list-decoding algorithm succeeds for these x's.

20.3 Multiplication code for B

Definition 4 *For* $B : \mathbf{Z}_N \to \{0,1\}$ *a predicate, define the* multiplication code *for* B, $C^B : \mathbf{Z}_N \to \{0,1\}^{\mathbf{Z}_N}$ *by*

$$C^B(x)_j = B(xj \mod N) \ . \tag{4}$$

There are two steps now for showing a predicate B is hard-core for p. First we will show that C^B is accessible for RSA and EXP. Then we show that C^B is an error-correcting code. Interestingly, the first part of the proof does not depend on the predicate B. The second part of the proof does not depend on the permutation RSA or EXP.

20.4 C^B is accessible for RSA, EXP

Lemma 2 $\forall B$, C^B *is accessible with respect to* RSA *and* EXP.

PROOF: For RSA, we want $A(x^e, j) = y^e$ such that $B(y) = C(x)_j = B(xj)$. Let $y = xj$, then $y^e = x^e j^e$ is computable. Over j, the distribution on y is uniform provided x is relatively prime to N. Therefore the distribution is $\sim \frac{1}{\sqrt{N}}$-far from uniform.

For EXP, we want $A(g^x \mod p, j) = g^y \mod p$, with $B(y) = C(x)_j = B(xj \mod p-1)$. Let $y = xj \mod p-1$, then $g^y = (g^x)^j \mod p$ is computable, and the distribution on y is close to uniform. $\square$

20.5 Predicates for which the multiplication code is list-decodable

Definition 5 $B : \mathbf{Z}_N \to \{0,1\}$ *is a* basic t-segment *predicate if in*

$$B(0), B(1), B(2), \ldots, B(N-1), B(0) \ , \tag{5}$$

there are $\leq t$ *changes of values.* B *is* t-segment *if for some invertible* $a \in \mathbf{Z}_N^*$, $B(xa)$ *is basic* t*-segment.*

For example, the most-significant bit, $\text{msb}(x)$, defined by

$$\text{msb}(x) = \begin{cases} 1 & \text{if } x \geq \lceil N/2 \rceil \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

is a basic 2-segment predicate. (Moreover, msb was previously known to be a hard-core predicate for RSA and EXP.)

As another example, if N is odd, then the least significant bit, $\text{lsb}(x) = x \mod 2$, is a 2-segment predicate. Indeed, $\text{msb}(x) = \text{lsb}(2x)$ – equivalently, $\text{lsb}(x) = \text{msb}(x/2) = \text{msb}(x\frac{N+1}{2})$. For if $x = 2y + b$ with $b \in \{0,1\}$ and $0 \leq y < \frac{N-b}{2}$, then

$$x\tfrac{N+1}{2} = (N+1)y + \tfrac{N+1}{2}b = \lceil \tfrac{N}{2} \rceil b + y \ . \tag{7}$$

Notice that EXP leaks the least significant bit (whether or not $g^x \mod p$ is a quadratic residue). The above argument fails since $\text{lsb}(2x) = 0$ in the domain $\mathbf{Z}_{p-1}$ since $p-1$ is even; 2 is not invertible.

Theorem 3 (Main) *Let $B : \mathbf{Z}_N \to \{0,1\}$ be a balanced, t-segment predicate. Then there is a list-decoding algorithm that given t and ϵ, and oracle access to $g : \mathbf{Z}_N \to \{0,1\}$ having agreement $\frac{1}{2}+\epsilon$ with $C^B(x)$, runs in time* $\text{poly}(\log N, t, 1/\epsilon)$ *and outputs a list that with high probability contains x.*

(By "balanced," we mean that B has at most a constant more, or fewer, zeros than ones. This condition, and the balance condition in Theorem 5 below, can be slightly weakened.)

To prove this theorem, recall the Fourier expansion of $f : \mathbf{Z}_N \to \mathbf{C}$: $f(x) = \sum_{a \in \mathbf{Z}_N} \hat{f}_a \chi_a(x)$, where $\chi_a(x) = \omega^{ax}$, $\omega = e^{2\pi i/N}$. Defining an inner product $f \cdot g = \frac{1}{N}\sum_x f(x)\overline{g(x)}$, the χ_a form an orthonormal basis, so $f \cdot g = \sum_a \hat{f}_a \overline{\hat{g}_a}$. If $f : \mathbf{Z}_N \to \{-1,1\}$, then $\sum_a |\hat{f}_a|^2 = 1$.

Now $C^B(x) \in \{0,1\}^{\mathbf{Z}_N}$, and we think of this encoding as $C^x : \mathbf{Z}_N \to \{-1,1\}$. Also, take $B : \mathbf{Z}_N \to \{-1,1\}$.

Definition 6 *$f : \mathbf{Z}_N \to \mathbf{C}$ is* concentrated *if $\forall\, \epsilon$, $\exists\, S$ with $|S| = \text{poly}(1/\epsilon)$ and $\sum_{a \in S} |\hat{f}_a|^2 \geq 1-\epsilon$.*

Theorem 4 (Concentration) *If B is a basic t-segment predicate, then for every x, C^x is concentrated (with $|S| = O(t/\epsilon^2)$).*

Theorem 5 (Agreement) *Assume $f, g : \mathbf{Z}_N \to \{-1,1\}$, f balanced and concentrated, agree on $\geq \frac{1}{2}+\epsilon$ fraction of inputs. Then there is $\tau = \text{poly}(\epsilon, 1/\log N)$ and $a \neq 0$ such that*

$$|\hat{f}_a|, |\hat{g}_a| \geq \tau \quad . \tag{8}$$

Theorem 6 (Learning) *There is an algorithm that given $g : \mathbf{Z}_N \to \{-1,1\}$ and $\epsilon > 0$, runs in time* $\text{poly}(\log N, 1/\epsilon)$ *and finds a list including all $a \in \mathbf{Z}_N$ with $|\hat{g}_a| \geq \epsilon$.*

Theorem 7 (Recovery) *If B is t-segment, C^B its multiplication code, there is an algorithm that given $a, \epsilon > 0$ runs in time* $\text{poly}(t, 1/\epsilon, \log N)$ *and finds a list including all $x \in \mathbf{Z}_N$ with $|\widehat{C^x}_a| \geq \epsilon$.*

Proof of main theorem: Given g, let τ be the constant of Theorem 5. Use the learning algorithm to find all a with $|\hat{g}_a| \geq \tau$. For each of them, use the recovery algorithm to find all y with $|\widehat{C^y}_a| \geq \tau$. In particular, by Theorem 5, the encoded x will be in this list. $\square$

20.6 Hadamard code

It is worth comparing the above framework to the Goldreich-Levin result for the permutation $x.r \mapsto p(x).r$. The Hadamard code is the multiplication code for the parity predicate, over the ring $\mathbf{Z}_2^{\times n}$. Codewords for the Hadamard code have exactly one Fourier coefficient 1 for the Fourier transform over $\mathbf{Z}_2^{\times n}$, and all others zero – so are certainly concentrated.

References

[AGS] Adi Akavia, Shafi Goldwasser, Samual Safra. Proving hard-code predicates using list-decoding. FOCS 2003.