CRYPTOGRAPHIC PRIMITIVES BASED ON HARD LEARNING PROBLEMS: BLUM, FURST, KEARNS, LIPTON

VISHVAS VASUKI

1. Depth with which this was read, progress

4 hours (10 - 3 pages).

2. Problems

What is the precise correspondence between hard to learn C and cryptographic primitives? Do representation independent hardness of learning results require cryptographic assumptions?

How to modify learning definitions to make hardness of learning assumption equivalent to a cryptographic assumption? How to use such assumptions to make cryptographic primitives?

3. Motivating real life scenarios

4. The Model

4.1. **Terms and Variables used.** Expansion of a pseudorandom bit generator.

F_n : functions over n-dim hypercube. $F = \cup F_n$. Representation scheme: (R_n, E_n) where every $s \in R_n$ is r(n) sized representation of $E_n(s) \in F_n$. $(R, E) = \cup \{(R_n, E_n)\}$.

Distribution over F_n or $R_n : P_n$. Their ensemble is P. Distribution over $\{0,1\}^n$: D_n. Their ensemble is D.

5. Results, methods and ideas

5.1. **The average case model of learning.** Modify learning definitions to make hardness of learning assumption equivalent to a cryptographic assumption.

5.1.1. *Motivation.* Take a hard to learn C. There may be some alg A which learns all but a small scattered fraction of C. If hardness of learning C were a cryptographic assumption, A would be a good attacker. So, use average case model of learning.

5.2. **Learnability and Cryptographic results.**

5.2.1. *Strong correspondence between hardness of learning and cryptography.*

5.2.2. *Generic transformation of C hard to weakly learn in the average case model into cryptographically secure pseudorandom bit generator.* Ckt depth depends on ckt depth of c in C and ckt required for generating hard distribution. Evidence that representation independent hardness of learning results require cryptographic assumptions. Is this truly the case?

If C hard to weakly learn in the average case model, even with mq, get pseudorandom bit generator with better expansion.

5.2.3. *If C hard to strongly learn in the average case model, get one way circuits.* Parallelism preserved: Faster one way functions imply faster key sharing protocols.

5.2.4. *If C hard to weakly learn in the average case model, get CPA secure private key crypto system.*

5.2.5. *Simpler construction of pseudorandom generator using the learning parity with noise assumption.*

6. Assumptions

7. What could have been tried to yield equivalent results

8. Open problems

9. Interesting facts and results from elsewhere

Pseudorandom bit generators can be turned into one way functions; and vice versa. But, sometimes, these don't preserve the circuit depth. How is this done?

The learning parities with noise assumption was used by Vaikuntanathan et al to build the first IBE system which does not require bilinear maps.

10. Comments on writing style

11. Questions