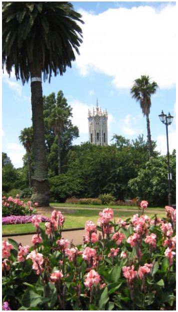
Open problems in applications of Fourier learning to the Diffie-Hellman problem in finite fields

Steven Galbraith

Mathematics Department, University of Auckland



Outline

- Hardcore bits
- Goldreich-Levin algorithm
- Decoding linear codes
- Fourier learning
- Applications revisited
- Open questions

Big thanks to Barak Shani and Joel Laity.

Some papers on elliptic curve bit security

- Dan Boneh and Igor Shparlinski. On the unpredictability of bits of elliptic curve Diffie- Hellman scheme. CRYPTO 2001.
- David Jao, Dimitar Jetchev and Ramarathnam Venkatesan: On the Bits of Elliptic Curve Diffie-Hellman Keys. INDOCRYPT 2007.
- Dimitar Jetchev and Ramarathnam Venkatesan: Bits Security of the Elliptic Curve Diffie-Hellman Secret Keys. CRYPTO 2008.

Applications of Fourier Learning in Cryptography

- Daniel Bleichenbacher. On The Generation of One-Time Keys in DL Signature Schemes. Talk, 2000.
- Adi Akavia, Shafi Goldwasser and Shmuel Safra: Proving Hard-Core Predicates Using List Decoding. FOCS 2003.
- Adi Akavia: Solving Hidden Number Problem with One Bit Oracle and Advice. CRYPTO 2009.
- Paz Morillo and Carla Ràfols: The Security of All Bits Using List Decoding. PKC 2009.
- Daniele Micciancio and Petros Mol: Pseudorandom Knapsacks and the Sample Complexity of LWE Search-to-Decision Reductions. CRYPTO 2011.

Applications of Fourier Learning in Cryptography

- Alexandre Duc and Dimitar Jetchev: Hardness of Computing Individual Bits for One-Way Functions on Elliptic Curves. CRYPTO 2012.
- Nelly Fazio, Rosario Gennaro, Irippuge Milinda Perera and William E. Skeith III: Hard-Core Predicates for a Diffie-Hellman Problem over Finite Fields. CRYPTO (2) 2013.
- Elke De Mulder, Michael Hutter, Mark E. Marson and Peter Pearson. Using Bleichenbacher's Solution to the Hidden Number Problem to Attack Nonce Leaks in 384-Bit ECDSA, CHES 2013.
- Diego F. Aranha, Pierre-Alain Fouque, Benot Grard, Jean-Gabriel Kammerer, Mehdi Tibouchi and Jean-Christophe Zapalowicz: GLV/GLS Decomposition, Power Analysis, and Attacks on ECDSA Signatures with Single-Bit Nonce Bias. Asiacrypt 2014.

Can these works tell us anything interesting about the DLP in finite fields?

Can these works tell us anything interesting about the DLP in finite fields?

Executive Summary:

Can these works tell us anything interesting about the DLP in finite fields?

Executive Summary:

No

So what is this all about?

Hardcore bits

- Let $f : X \to Y$ be a one-way function.
- This means: Given x one can efficiently compute $y = f(x)$, but given y it is computationally infeasible to find $x \in X$ such that $f(x) = y$.
- By definition, given $f(x)$ one cannot efficiently deduce x. But, given $f(x)$ it might be possible to compute some **partial information** about x.
- Let $X = \{0, 1\}^n$ and $\underline{x} = (x_1, \ldots, x_n) \in X$. Define $\mathrm{bit}_i(\underline{x}) = x_i$.

Hardcore bits

- Let $f : X \to Y$ be a one-way function.
- We say the i-th bit is **easy** for f if, given $f(\underline{x})$, one can compute the value $\mathrm{bit}_i(\underline{x})$ with probability significantly better than guessing.
  (I assume the values $\underline{x}$ are sampled uniformly in the game.)
- It is immediate that a one-way function has many bits that are not easy.
  We call such bits **hardcore bits** for f.
- The problem is to prove that a specific bit is hardcore.

Proof technique

- To show $\text{bit}_i(x)$ is hardcore one argues as follows:
- Suppose one has an algorithm/oracle that on input any $y \in Y$ outputs with high probability the correct value of $\text{bit}_i(x)$ where $y = f(x)$.
- Suppose one is also given a challenge $y^* = f(x^*)$.
- Then one constructs an efficient algorithm that computes x^* by making oracle queries to the algorithm.

Proof technique

- When f is "algebraic" and the oracle is always correct then this is easy.
- For example, suppose $f(x) = g^x \pmod{p}$ where g has odd prime order r.
- Let O be an oracle such that $O(y)$ returns $\mathrm{bit}_0(x)$.
- Let $y^* = g^{x^*}$ be the challenge.
- Calling $O(y^*)$ gives x_0^*.
- Now set

$$y = (y^* g^{-x_0^*})^{2^{-1} \pmod{r}}$$

- Calling $O(y)$ gives x_1^*.
- The process repeats in the obvious way.

Hardcore bits

- This approach generalises to functions other than bits.
- Let $g : X \to Z$ be some function.
- Then one can talk about whether $g(x)$ is hardcore.
- For example $g(x_1, \ldots, x_n) = (x_1, \ldots, x_k)$ for some $1 < k < n$.
- One is arguing that it is hard to compute the first k bits of x.
- This is a weaker notion: It might be hard to compute the first k bits of x, but that does not imply it is hard to compute the first bit of x.

Ancient history

- The big challenge is to handle oracles that are not correct all the time.
- First case of interest was hardcore bits for RSA.
- Goldwasser, Micali and Tong (FOCS 1982): least significant bit of RSA is hardcore if oracle is correct with probability $1 - 1/\log_2(N)$.
  In other words, make about $\log_2(N)$ oracle queries and only one of them is wrong.
- Ben Or, Chor and Shamir (STOC 1983): least significant bit of RSA is hardcore if oracle is correct with probability $3/4 + \epsilon$.
- Alexi, Chor, Goldreich and Schnorr (1988): handles oracle that is correct with probability $1/2 + \epsilon$.
- None of these papers discuss Fourier analysis.

A general solution

- Goldreich, Levin (1989) gave a construction to derive a hardcore bit for any one-way function.
- Idea: Let $X = \mathbb{F}_2^n$. Given $\underline{x} \in X$ one chooses $\underline{t} \in X$ and sends $(\underline{t}, f(\underline{x}))$.
- The hardcore bit is

$$\underline{t} \cdot \underline{x} = \sum_{j=1}^{n} t_j x_j.$$

- Idea: Let O be an oracle that, on input $f(\underline{x})$, outputs $\underline{t} \cdot \underline{x}$ non-negligibly better than guessing.
  Using O and choosing $\underline{t}$ one can compute $\underline{x}$.
- It was pointed out by Rackoff and Wigderson that the Goldreich-Levin algorithm is based on the Walsh transform (fourier analysis in the group $\mathbb{F}_2^n$).

Elementary approach to Goldreich-Levin

- Let O be an oracle that, for a secret value $\underline{x}$, on input $\underline{t}$ outputs $\underline{t} \cdot \underline{x}$ non-negligibly better than guessing.
- If there are no errors: query $O(\underline{e}_i)$ for unit vectors $\underline{e}_i$.
- Basic trick: Choose random $\underline{a}$ and query $O(\underline{a} + \underline{e}_i) - O(\underline{a})$.
- If both oracle outputs correct then have

$$(\underline{a} + \underline{e}_i) \cdot \underline{x} - \underline{a} \cdot \underline{x} \;=\; x_i.$$

- Repeat for many random $\underline{a}$ and take majority vote.
- **Note:** It is essential to be able to choose the inputs to O.

Connection with decoding linear codes

- We are getting a lot of values $\underline{t} \cdot \underline{x}$, for various $\underline{t} \in X$, some of them with errors.
- Putting all the rows $\underline{t}$ together as an $m \times n$ matrix we have measurements $T\underline{x} + \underline{e}$ where $\underline{e}$ is a length m column vector of low weight.
- Hence, can re-phrase computing $\underline{x}$ in terms of T being the "generator matrix" of a code and $T\underline{x} + \underline{e}$ being the received "code word".

Connection with decoding linear codes

- We are getting a lot of values $\underline{t} \cdot \underline{x}$, for various $\underline{t} \in X$, some of them with errors.
- Putting all the rows $\underline{t}$ together as an $m \times n$ matrix we have measurements $T\underline{x} + \underline{e}$ where $\underline{e}$ is a length m column vector of low weight.
- Hence, can re-phrase computing $\underline{x}$ in terms of T being the "generator matrix" of a code and $T\underline{x} + \underline{e}$ being the received "code word".
- This is of course a bit of a cheat: We are essentially choosing the generator matrix to have a special structure.

Fourier Analysis on Finite Groups

- Consider $G = \mathbb{F}_2^n$, a finite additive group of order 2^n.
- The set of functions $f : G \to \mathbb{C}$ is a $\mathbb{C}$-vector space of dimension 2^n.
- There is an inner product

$$\langle f, g \rangle \;=\; \tfrac{1}{2^n} \sum_{\underline{x} \in G} f(\underline{x}) \overline{g(\underline{x})}$$

- An orthonormal basis for this set of functions is

$$\chi_{\underline{a}}(\underline{x}) = (-1)^{\underline{a} \cdot \underline{x}}$$

where $\underline{a}$ runs over all elements of $\mathbb{F}_2^n$.

Fourier analysis on finite groups

- Let $f : G \to \mathbb{C}$ be given, $G = \mathbb{F}_2^n$.
- Then f has a Fourier expansion

$$f(\underline{x}) \;=\; \sum_{\underline{a} \in \mathbb{F}_2^n} \hat{f}(\underline{a}) \chi_{\underline{a}}(\underline{x})$$

  where the Fourier coefficients are $\hat{f}(\underline{a}) = \langle f, \chi_{\underline{a}} \rangle$.
- Parseval's identity: $\langle f, f \rangle = \sum_{\underline{a} \in G} \hat{f}(\underline{a})^2$.
- We call a character $\chi_{\underline{a}}$ **heavy** if $|\hat{f}(\underline{a})|$ is relatively large with respect to $\langle f, f \rangle$.
- Parseval implies there cannot be many heavy Fourier coefficients.
- $f(\underline{x})$ is called **concentrated** if it has some heavy Fourier coefficients.

Lemma

- Let $f : \mathbb{F}_2^n \to \{1, -1\}$ be such that

$$f(\underline{x}) = (-1)^{\underline{x}\cdot\underline{s}} = \chi_{\underline{s}}(\underline{x})$$

for all $\underline{x} \in X \subseteq \mathbb{F}_2^n$, and

$$f(\underline{x}) = (-1)^{\underline{x}\cdot\underline{s}+1} = -\chi_{\underline{s}}(\underline{x})$$

for all $\underline{x} \in \overline{X} = \mathbb{F}_2^n \setminus X$.
- Let $|X| = \delta 2^n$.
- Then $\langle f, f \rangle = 1$ and $\hat{f}(\underline{s}) = 2\delta - 1$.

Fourier approach to Goldreich-Levin

- Let $G = \mathbb{F}_2^n$ and fix $\underline{s} \in G$.
- Let $f(\underline{t}) : G \to \{-1, 1\}$ be such that on $1/2 + \delta$ of the inputs $\underline{t} \in G$ we have $f(\underline{t}) = (-1)^{\underline{t} \cdot \underline{s}}$.
- Consider the Fourier series for $f(\underline{t})$.
- By the previous Lemma, $f(\underline{t})$ is concentrated and $\chi_{\underline{s}}$ is a heavy Fourier character.
  In other words, the coefficient $|\hat{f}(\underline{s})|$ is large.
- One extends the algorithmic ideas from learning one secret $\underline{s}$ to learning all the heavy Fourier characters.

List decoding connection

- Suppose $f(\underline{x})$ is a function on $G = \mathbb{F}_2^n$ which has several heavy characters $\underline{s}_1, \ldots, \underline{s}_k$.
- It means $f(\underline{x})$ and $(-1)^{\underline{x} \cdot \underline{s}_j}$ agree on a large set of inputs for each $1 \leq j \leq k$.
- We can write T again for the matrix corresponding to the function queries we will make, and represent the outputs of f (turned back from $\{-1, 1\}$ to $\{0, 1\}$) as a codeword.
- An algorithm that computes a list of heavy coefficients $\underline{s}_1, \ldots, \underline{s}_k$ can be viewed as a list decoding algorithm.

Further work

- There are general algorithms to compute all heavy Fourier coefficients of any concentrated function on a finite abelian group G.
- Best paper to read to get the main ideas is Kushilevitz and Mansour (STOC 1991).
- There are improved algrithms, see recent survey paper by Gilbert, Indyk, Iwen and Schmidt.
- All these works require chosen queries to the function.

Bit security of CDH

- Given $g, g^a, g^b \in \mathbb{F}_p^*$ want to know what bits of g^{ab} are secure.
- So suppose one has an oracle $O(g, g^a, g^b)$ that outputs some bits of g^{ab}.
- We want to use O to compute all of $s = g^{ab}$.
- Idea is to choose random r and call $O(g, g^a, g^b g^r)$, which gives bits of $g^{a(b+r)} = s(g^a)^r$.
- Hidden number problem: Fix $s \in \mathbb{F}_p^*$ and let O be an oracle such that $O(t) = \mathrm{LSB}(st)$. Goal is to compute s given access to O.

Boneh-Venkatesan

- Hidden number problem: Given $(t, \text{bits}(ts))$ to compute $s \in \mathbb{F}_p^*$.
- BV consider oracle that computes the $\sqrt{\log(p)}$ most significant bits of the Diffie-Hellman secret $s = g^{ab}$.
- Oracle is always correct.
- Multipliers t chosen randomly and non-adaptively.
- BV use lattice method.
- Lots of following work.
- Challenge: one bit, unreliable oracle.

Fourier analysis in $\mathbb{F}_p^*$

- Let $G = \mathbb{F}_p^*$ and consider functions $f : G \to \mathbb{C}$.
- Let $\zeta = \exp(2\pi i/p)$.
- For $a \in \{1, \ldots, p-1\}$ define the character

$$\chi_a(x) = \zeta^{ax}.$$

- Then, for $a, b \in \{1, \ldots, p-1\}$

$$\langle \chi_a, \chi_b \rangle = \frac{1}{p} \sum_{x=1}^{p-1} \chi_a(x) \overline{\chi_b(x)} = \frac{1}{p} \sum_{x=1}^{p-1} \zeta^{(a-b)x}.$$

This is 0 if $a \neq b$ and 1 if $a = b$.

Application: Bit security of CDH

- Given $g, g^a, g^b \in \mathbb{F}_p^*$ want to prove that a single bit of g^{ab} is secure.
- Akavia et al considered function $f(x) : \mathbb{F}_p \to \{-1, 1\}$ by $f(x) = (-1)^{\mathrm{LSB}(x)}$.
  One can show that this is concentrated.
  [Also $(-1)^{\mathrm{MSB}(x)}$.]
- Big result from CRYPTO 2009: "Solving Hidden Number Problem with One Bit Oracle and Advice".
- Idea: Translation of Fourier coefficients by s.
- Recall that we choose random r and call $O(g, g^a, g^b g^r)$ to get a bit of $g^{a(b+r)} = s(g^a)^r$.

Applications to elliptic curves

- Would like to prove certain bits are hardcore for ECDH.
- Given P, aP, bP want to show that if can compute some bit of abP then can compute all abP.
- Boneh and Shparlinski developed an approach.
- Suppose O is an oracle that takes (E, P, aP, bP) such that $P \in E$ and computes the most significant bits of the x-coordinate of abP.

Boneh and Shparlinski

- Trick is to consider isomorphism $\phi : E \to E'$ of Weierstrass curves given by
$$\phi(x, y) = (\lambda^2 x, \lambda^3 y).$$
- Then call $O(E', \phi(P), \phi(aP), \phi(bP))$.
- The secret is $s = x(abP)$ and, for chosen λ, get bits of $\lambda^2 s$, which is more-or-less back to hidden number problem with chosen multipliers.
- Boneh and Shparlinski used lattice methods.
- Duc and Jetchev used Fourier methods.

Elliptic curve Diffie-Hellman

- Changing Weierstrass models does not actually prove a hardcore bit for a fixed representation.
- I call it the "BS" trick.
- Hence, the problem of hardcore bits for ECDH is still open.
- Any ideas?

Obstruction

- "Beurling-Helson theorem".
- $f(x)$ concentrated and $f(g(x))$ concentrated implies that $g(x)$ is an affine map.
- If $g(x)$ comes from elliptic curve operations then it is a non-trivial rational function.
- Hence, elliptic curve operations do not preserve concentrated.

CDH in non-prime finite fields

- Fazio, Gennaro, Perera and Skeith (CRYPTO 2013) showed bit security for CDH in $\mathbb{F}_{p^2}$.
- Their model involves field isomorphisms (a version of the "BS trick").
- The paper "The Multivariate Hidden Number Problem" (ICITS 2015) written with my student Barak Shani treats general fields $\mathbb{F}_{p^n}$.
- Also uses a "BS trick".
- It is still open to prove bit security of Diffie-Hellman in finite fields.

Open Questions

- Do these ideas have anything to do with DLP?
- Can one actually prove bit security results for single bits in realistic models?
- Are Fourier learning algorithms optimal from a concrete point of view?

Thank you for your attention

See you at Asiacrypt!