Solving Hidden Number Problem
with One Bit Oracle and Advice

Adi Akavia*

[1] Institute for Advanced Study, Princeton NJ 08540
[2] DIMACS, Rutgers University, Piscataway, NJ 08854

Abstract. In the *Hidden Number Problem (HNP)*, the goal is to find a hidden number s, when given p, g and access to an oracle that on query a returns the k most significant bits of $s \cdot g^a \bmod p$.

We present an algorithm solving HNP, when given an advice depending only on p and g; the running time and advice length are polynomial in $\log p$. This algorithm improves over prior HNP algorithms in achieving: (1) optimal number of bits $k \geq 1$ (compared with $k \geq \Omega(\log\log p)$); (2) robustness to random noise; and (3) handling a wide family of predicates on top of the most significant bit.

As a central tool we present an algorithm that, given oracle access to a function f over $\mathbb{Z}_N$, outputs all the significant Fourier coefficients of f (i.e., those occupying, say, at least 1% of the energy). This algorithm improves over prior works in being:

- *Local.* Its running time is polynomial in $\log N$ and $L_1(\widehat{f})$ (for $L_1(\widehat{f})$ the sum of f's Fourier coefficients, in absolute value).
- *Universal.* For any N, t, the *same* oracle queries are asked for *all* functions f over $\mathbb{Z}_N$ s.t. $L_1(\widehat{f}) \leq t$.
- *Robust.* The algorithm succeeds with high probability even if the oracle to f is corrupted by random noise.

1 Introduction

The *Hidden Number Problem (HNP)* was introduced by Boneh and Venkatesan [4] in the context of proving bit security for the Diffie-Hellman function. In HNP, for p a prime, and g a generator of $\mathbb{Z}_p^*$, the goal is to find a hidden number $s \in \mathbb{Z}_p^*$, when given p, g and oracle access to the function

$$P_{p,s,k}(a) \overset{def}{=} MSB_{p,k}(s \cdot g^a \bmod p)$$

mapping each $a \in 1, \ldots, p$ to the k most significant bits in the binary representation of $s \cdot g^a \bmod p$.

Boneh-Venkatesan [4] gave an algorithm solving HNP for any $k \geq \sqrt{\log p} + \log\log p$ in running time polynomial in $\log p$ (aka, *efficient*). Subsequently, Boneh-Venkatesan [5] gave an efficient algorithm solving HNP for $k \geq \Omega(\log\log p)$ provided the algorithm is given a short *advice* depending only on p and g (and not on s). Extensions to the case g is not a generator are given in [8, 14, 15].

* This research was supported in part by NSF grant CCF-0514167, by NSF grant CCF-0832797, and by Israel Science Foundation 700/08. Email: akavia@ias.edu

1.1 New Result: Solving HNP with One Bit Oracle & Advice

We present an efficient algorithm solving HNP for any $k \geq 1$, provided the algorithm is given a short advice depending only on p and g (and not on s).

Furthermore, our algorithm handles:

- *Random noise.* With high probability, our algorithm finds s even if the oracle answers are flipped independently at random with sufficiently small probability $\varepsilon > 0$. (Success probability is taken over the noise.)
- *Concentrated predicates.* Our algorithm finds s even when oracle access is to the function
$$P_{p,s}(a) \overset{def}{=} P_p(s \cdot g^a \bmod p)$$
where $\mathcal{P} = \{P_p\}$ is any family of "concentrated" predicates. We say that $\mathcal{P}$ is *concentrated* if
$$\exists c, \delta \text{ s.t. } \forall P_p \in \mathcal{P}, L_1(\widehat{P_p}) \leq (\log p)^c \text{ and } \mathsf{maj}(P_p) \leq 1 - \delta$$
for $L_1(\widehat{P_p}) \overset{def}{=} \sum_\alpha \left|\widehat{P_p}(\alpha)\right|$ the sum of Fourier coefficients, and $\mathsf{maj}(P_p) \overset{def}{=} \max_{b=0,1} \Pr_{a \in \mathbb{Z}_p}[P_p(a) = b]$ the frequency of the most common value.

Noise is tolerated up to $\varepsilon = c'\tau(\mathcal{P})$ for any $c' < 1$ and for any $\tau(\mathcal{P})$ a lower bound on the maximum squared magnitude of the (non-trivial) Fourier coefficients of predicates $P_p \in \mathcal{P}$. In particular, for $\mathcal{P}$ the most significant bit, $\varepsilon = O(1)$.[3]

As a corollary of our algorithm for HNP, we obtain bit security results for Diffie-Hellman related functions.

Our result improves on prior HNP algorithms (and the corresponding bit security results) in achieving:

1. Optimal number of bits $k \geq 1$ (rather than $k \geq \Omega(\log \log p)$);
2. Robustness to ε-random noise for substantial ε (e.g., ε is $O(1)$ rather than $O(1/\log p)$ for $\mathcal{P} = \mathcal{MSB}_k$ the k most significant bits); and
3. Handling the wide family of concentrated predicates (rather than only $\mathcal{MSB}_k$).

1.2 New Tool: Universally Finding Significant Fourier Coefficients

As a central tool we present an algorithm that finds the significant Fourier coefficients of a complex valued functions f over $\mathbb{Z}_p$, when given oracle access to f (aka, SFT algorithm).

Indexing Fourier coefficients by elements α in $\mathbb{Z}_p$, we say that α is τ*-significant* if its Fourier coefficient occupies at least τ-fraction of the energy
$$\left|\widehat{f}(\alpha)\right|^2 \geq \tau \sum_{\beta \in \mathbb{Z}_p} \left|\widehat{f}(\beta)\right|^2 \quad .$$

Our SFT algorithm, given p, τ, t, and oracle access to a function f over $\mathbb{Z}_p$ s.t. $L_1(\widehat{f}) \leq t$, outputs all the τ-significant Fourier coefficients of f. Our SFT algorithm is:

[3] For $\mathcal{P}$ the $k \geq \Omega(\log \log p)$ most significant bits, prior works [5] tolerate adversarial noise corrupting up to $\varepsilon = O(1/\log p)$ fraction of the oracle values.

- *Local.* Its running time is polynomial in $\log p$, $1/\tau$ and t.
- *Universal.* For any p, τ and t, the *same* oracle queries are asked for *all* functions f over $\mathbb{Z}_p$ s.t. $L_1(\widehat{f}) \leq t$.
- *Robust.* With high probability, the algorithm succeeds even if the oracle to f is corrupted by random noise (probability is taken over the noise). Tolerated noise parameters are up to $\varepsilon = c\tau$ for any constant $c < 1$.

This improves over prior works in giving: (i) The first universal algorithm handling all functions f over $\mathbb{Z}_p$ (complexity scales with $L_1(\widehat{f})$). (ii) The first analysis proving robustness to noise in the context of universal SFT algorithms. We remark that these improvements are of independent interest in the context of sparse Fourier approximation, compressed sensing and sketching (cf. [3]).

Comparison to other SFT algorithms. For functions over the boolean hyper-cube $\mathbb{Z}_2^n$, Kushilevitz-Mansour (KM) gave a local universal SFT algorithm almost two decades ago [12]. Our algorithm matches the KM benchmark for the case of functions over $\mathbb{Z}_p$ for any positive integer p.

For functions over $\mathbb{Z}_p$, prior SFT algorithms [6, 2, 7] are not universal. In concurrent works [10, 11] gave a universal SFT algorithm for a restricted class of functions over $\mathbb{Z}_p$: *compressible* or *Fourier sparse* functions.[4]

Noise is out of scope in the analysis of the universal algorithms [12, 10, 11].

These SFT algorithms [12, 6, 2, 7, 10, 11] are insufficient for our result solving HNP. Both universality as well as handling functions that are neither compressible nor Fourier sparse are crucial for our algorithm solving HNP. Robustness to noise leads to robustness when solving HNP.

1.3 Techniques Overview

In HNP the goal is to find a hidden number s when given p, g and oracle access to a function $P_{p,s}$. We reduce the HNP problem to the problem of the finding significant Fourier coefficients of a function f_s defined by

$$f_s(y) \stackrel{def}{=} P_{p,s}(DL_{p,g}(y))$$

for $DL_{p,g}(y)$, the discrete log of y, i.e., the $a \in \mathbb{Z}_{p-1}$ s.t. $y = g^a \bmod p$. We then find the significant Fourier coefficients of f_s using our universal SFT algorithm.

Universality is crucial. Finding the Fourier coefficients of f_s requires access to f_s. To read the values $f_s(y)$ on entries y it suffices to query $P_{p,s}$ on the *discrete-logs* $DL_{p,g}(y)$. With universal algorithms, access to all entries y read by the algorithm can be granted using an *advice* depending only on p. This is because universal algorithms read a *fixed* set of entries y for all the considered functions over $\mathbb{Z}_p$; implying that the discrete-logs $DL_{p,g}(y)$ for all read entries

[4] For g a function over $\mathbb{Z}_p$ and $c, c' > 0$ absolute constants (indep. of p), g is *compressible* if for all i, the i-th largest Fourier coefficient of g has magnitude at most $O(1/c^i)$; and g is *Fourier sparse* if it has at most $(\log p)^{c'}$ non-zero Fourier coefficients.

y can be provided via an advice depending only on p. In contrast, with non-universal algorithms, providing access to f_s is intractable (assuming computing discrete logs is intractable).

Achieving universality. We say that a set of queries $S \subseteq \mathbb{Z}_p$ is *good* if we can find the significant Fourier coefficients of all considered function over $\mathbb{Z}_p$ when reading only entries in S. We present a combinatorial condition on sets S, and prove that any set S satisfying this condition is good. Furthermore, we show that sets S satisfying the condition exists, and can be efficiently construction by a randomized algorithm. We remark that explicit constructions of such good sets are given in subsequent works [3].

The combinatorial condition is that $S = \cup_{\ell=0}^{\log p}(A - B_\ell)$ for A a small biased set and B_ℓ's that are "small biased on $[0..2^\ell]$"; where we say that B *has small bias on* I if Fourier coefficients of (the characteristic function of) B approximate the Fourier coefficients of (the characteristic function) of I.

We prove that such sets S are good in two parts. First, for functions with bounded $L_1(\widehat{f})$, we prove S is good using Fourier analysis. Second, for noise corrupted functions $f' = f + \eta$, we prove S is good by showing the algorithm behaves similarly on the noisy and non-noisy functions. The latter is needed, as the Fourier approach fails for noisy f' due to their typically huge $L_1(\widehat{f'}) \approx \sqrt{p}$.

Comparison to prior works. Prior algorithms solving HNP follow a lattice based approach dating back to [4], in which HNP is reduced to the problem of finding closest lattice vectors (CVP), and the latter is solved using LLL algorithm [13]. In comparison, we take a Fourier approach inspired by [2].

We compare the set of queries used in the different SFT algorithms.

In the universal SFT algorithm for functions over the boolean hypercube $\mathbb{Z}_2^n$ [12], the set of queries is constructed using small biased sets in $\mathbb{Z}_2^n$, and the proof is Fourier analysis based.

In the (non-universal) SFT algorithms for functions over $\mathbb{Z}_p$ [6, 2, 7], the set of queries must be freshly chosen for each given input function f. Their analysis proves success with high probability over the sampled set of queries using deviation from expectation bounds.

In the universal SFT algorithm for (restricted class of) functions over $\mathbb{Z}_p$ [10, 11], the set of queries is constructed using "K-majority k-strongly selective sets".

1.4 Paper Organization

The rest of this paper is organized as follows. In section 2 we summarize preliminary terminology, notations and facts. In section 3 we present our algorithm solving HNP with advice. In section 4 we present our universal SFT algorithm. In section 5 we discuss bit security implications.

2 Preliminaries

In this section we summarize preliminary terminology, notations and facts.

Let $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{R}$ and $\mathbb{C}$ denote the natural, integer, real and complex numbers respectively. Let $\mathbb{P}$ denote the set of all primes. Let $\mathbb{Z}_N$ and $\mathbb{Z}_N^*$ denote the additive and the multiplicative groups of integers modulo N. We identify the elements of $\mathbb{Z}_N$ with integers in $0,\dots,N-1$, and denote $\mathsf{abs}(\alpha) = \min\{\alpha, N-\alpha\}$ for all $\alpha \in \mathbb{Z}_N$. Let $\mathbb{B}_r \stackrel{def}{=} \{z \in \mathbb{C} \mid |z| \leq r\}$ denote the complex ball of radius r.

2.1 Fourier Transform

We give definitions and properties for normed spaces and Fourier transform.

Inner product, norms, convolution. The *inner product* of complex valued functions f, g over a domain G is $\langle f, g\rangle \stackrel{def}{=} \frac{1}{|G|}\sum_{x\in G} f(x)\overline{g(x)}$. Denote the normalized ℓ_2 norm of f by $\|f\|_2 \stackrel{def}{=} \sqrt{\langle f, f\rangle}$, its ℓ_∞ norm by $\|f\|_\infty \stackrel{def}{=} \max\{|f(x)| \mid x \in G\}$, and its un-normalized L_1-norm by $L_1(f) \stackrel{def}{=} \sum_{x\in G}|f(x)|$. The *convolution* of f and g is the function $f * g\colon G \to \mathbb{C}$ defined by $f * g(x) \stackrel{def}{=} \frac{1}{|G|}\sum_{y\in G} f(y)\overline{g(x-y)}$.

Characters and Fourier transform. The *characters* of $\mathbb{Z}_N$ are the functions $\chi_\alpha\colon \mathbb{Z}_N \to \mathbb{C}$, $\alpha \in \mathbb{Z}_N$, defined by $\chi_\alpha(x) \stackrel{def}{=} e^{2\pi i\alpha x/N}$. The *Fourier transform* of a complex valued function f over $\mathbb{Z}_N$ is the function $\widehat{f}\colon \mathbb{Z}_N \to \mathbb{C}$ defined by $\widehat{f}(\alpha) \stackrel{def}{=} \langle f, \chi_\alpha\rangle$. For any $\alpha \in \mathbb{Z}_N$ and $\tau \in [0,1]$, we say that α is a τ-*significant* Fourier coefficient iff $\left|\widehat{f}(\alpha)\right|^2 \geq \tau\|f\|_2^2$. Denote by $\mathsf{Heavy}_\tau(f)$ the set of all τ-significant Fourier coefficients of f.

A few useful properties of the Fourier transform follow.

Proposition 1. *For any $f, g\colon \mathbb{Z}_N \to \mathbb{C}$,*

1. *Parseval Identity:* $\frac{1}{N}\sum_{x\in\mathbb{Z}_N}|f(x)|^2 = \sum_\alpha \left|\widehat{f}(\alpha)\right|^2$.
2. *Convolution Theorem:* $\widehat{(f*g)}(\alpha) = \widehat{f}(\alpha)\cdot\widehat{g}(\alpha)$.
3. *Phase Shift: For any $\alpha_0 \in \mathbb{Z}_N$, if $g = f\cdot\chi_{-\alpha_0}$, then $\widehat{g}(\alpha) = \widehat{f}(\alpha-\alpha_0)$ (where subtraction is modulo N).*
4. *Scaling: For any $s \in \mathbb{Z}_N^*$, if $g(x) = f(sx)\ \forall x$, then $\widehat{g}(\alpha) = \widehat{f}(\alpha\cdot s^{-1})\ \forall\alpha$ (where multiplication and inverse are modulo N).*

Proof. Proof is standard, see [16]. □

Proposition 2. *Let $S_t(\alpha) \stackrel{def}{=} \frac{1}{t}\sum_{y=0}^{t-1}\chi_\alpha(y)$ for some $t \in [0..N-1]$. Then:*

1. $|S_t(\alpha)|^2 = \frac{1}{t^2}\frac{1-\cos(\frac{2\pi}{N}\alpha t)}{1-\cos(\frac{2\pi}{N}\alpha)}$
2. *Pass Band: $\forall\alpha \in \mathbb{Z}_N$ and $\gamma \in [0,1]$, if $\mathsf{abs}(\alpha) \leq \gamma\frac{N}{2t}$, then $|S_t(\alpha)|^2 > 1 - \frac{5}{6}\gamma^2$*
3. *Fast decreasing: $\forall\alpha \in \mathbb{Z}_N$, $|S_t(\alpha)|^2 < \frac{2}{3}\left(\frac{N/t}{\mathsf{abs}(\alpha)}\right)^2$*
4. *Fourier bounded: $\forall\alpha \in \mathbb{Z}_N$, $|S_t(\alpha)|^2 \leq 1$*

Proof. Recall that $\chi_\alpha(x) = \omega^{\alpha x}$ for $\omega = e^{i\frac{2\pi}{N}}$ a primitive root of unity of order N. By the formula for geometric sum $S_t(\alpha) = \frac{1}{t}\frac{\omega^{-\alpha t}-1}{\omega^{-\alpha}-1}$. Assigning $w^\beta = \cos(2\pi\beta/N)+i\sin(2\pi\beta/N)$ for $\beta = \alpha t$ in the numerator and $\beta = \alpha$ in the denominator and using standard trigonometric identities, we conclude that $|S_t(\alpha)|^2 = \frac{1}{t^2}\frac{1-\cos(\frac{2\pi}{N}\alpha t)}{1-\cos(\frac{2\pi}{N}\alpha)}$. The upper and lower bounds on S_t are obtained using the Taylor approximation for the cosine function: $1 - \frac{\theta^2}{2!} \leq \cos(\theta) \leq 1 - \frac{\theta^2}{2!} + \frac{\theta^4}{4!}$. Details appear in [2, 1]. □

2.2 Chernoff/Hoeffding Tail inequality

The Chernoff/Hoeffding bound on the deviation from expectation of sums of independent random variables follows.

Proposition 3 (Chernoff/Hoeffding Bound [9]). *Let $X_1, \ldots, X_t$ be independent random variables of expectations $\mu_1, \ldots, \mu_y$ and bounded values $|X_i| \leq M$. Then, $\forall\eta > 0$,* $\Pr[\left|\frac{1}{t}\sum_{i=1}^t X_i - \frac{1}{t}\sum_{i=1}^t \mu_i\right| \geq \eta] \leq 2 \cdot exp\left(-\frac{2t\eta^2}{M^2}\right)$.

2.3 Noise Models

We say that η is an *ε-random noise* if its values $\eta(a)$, $a \in \mathbb{Z}_p$, are chosen independently at random from distributions of expected absolute values at most $\mathbb{E}[|\eta(a)|] \leq \varepsilon$.

We focus on additive noise η corrupting functions f to a function $f' = f + \eta$. Without loss of generality, f and η accept values in the balls $\mathbb{B}_1$, $\mathbb{B}_2$ respectively.

3 Solving Hidden Number Problem with Advice

In this section we present our algorithm solving with advice $\mathrm{HNP}^{\mathcal{P},\varepsilon}$.

Fix a family of functions $\mathcal{P} = \left\{P_p\colon \mathbb{Z}_p^* \to \mathbb{B}_1\right\}_{p\in\mathbb{P}}$ and a noise parameter ε.

Definition 1 (Hidden Number Problem). *In the* (extended) Hidden Number Problem $\mathrm{HNP}^{\mathcal{P},\varepsilon}$ *the goal is to find a hidden number $s \in \mathbb{Z}_p^*$, when given a prime p, a generator g of $\mathbb{Z}_p^*$, and oracle access to the function*

$$P'_{p,s}(a) \overset{def}{=} P_p(s \cdot g^a \bmod p) + \eta(a)$$

for η an ε-random noise.

Let ℓ, q, t be functions over $\mathbb{P}$. We say that an algorithm (ℓ, q, t)-*solves* $\mathrm{HNP}^{\mathcal{P},\varepsilon}$ if there is an advice $Adv_{p,g}$ depending only on p, g of length $|Adv_{p,g}| \leq \ell(p)$, such that the following holds. Given p, g, $Adv_{p,g}$, and oracle access to $P'_{p,s}$, the algorithm outputs s with probability at least $q(p)$; and its running time is at most $t(p)$. We say that the algorithms *solves with advice* $\mathrm{HNP}^{\mathcal{P},\varepsilon}$ if $1/q(p), \ell(p)$ and $t(p)$ are polynomial in $\log p$.

3.1 Solving with Advice $\mathrm{HNP}^{\mathcal{P},\varepsilon}$: Concentrated $\mathcal{P}$

We present an efficient algorithm solving with advice $\mathrm{HNP}^{\mathcal{P},\varepsilon}$ for concentrated $\mathcal{P}$. We remark that concentration defined here differ than concentration in [2].

Let M, τ and α be functions mapping indices $p \in \mathbb{P}$ into non-negative reals $M(p), \tau(p)$ and a non-zero element $\alpha(p) \in \mathbb{Z}_p$.

Definition 2 (Concentration). *$\mathcal{P}$ is* (M, τ, α)-concentrated *if for all $P_p \in \mathcal{P}$,*

$$L_1(\widehat{P_p}) \leq M(p) \quad \text{and} \quad \left|\widehat{P_p}(\alpha(p))\right|^2 \geq \tau(p).$$

$\mathcal{P}$ is concentrated *if $\exists c > 0$ s.t. $\forall p \in \mathbb{P}$, $M(p)$ and $1/\tau(p)$ are at most $(\log p)^c$.*

Let $\tau(\mathcal{P})$ denote a lower bound on the maximum weight $\left|\widehat{P_p}(\alpha)\right|^2$ of non-trivial Fourier coefficients $\alpha \neq 0$, for all $P_p \in \mathcal{P}$.

Theorem 1 ($\mathrm{HNP}^{\mathcal{P},\varepsilon}$). *For any concentrated $\mathcal{P}$ and $\varepsilon \leq c \cdot \tau(\mathcal{P})$ for $c < 1$, there exists an algorithm that solves with advice $HNP^{\mathcal{P},\varepsilon}$.*

Proof. Let m, τ, α be s.t. $\mathcal{P}$ is (M, τ, α)-concentrated. We present an algorithm that (ℓ, q, t)-solves $\mathrm{HNP}^{\mathcal{P},\varepsilon}$ for $q(p) \geq \Omega(\tau(p))$ and for $\ell(p), t(p)$ polynomial in $\log p$, $M(p)$ and $1/\tau(p)$. The advice we use is:

$$Adv_{p,g} \stackrel{def}{=} \{(x, DL_{p,g}(x))\}_{x \in S}$$

for $S \subseteq \mathbb{Z}_p$ a set of good queries for our universal SFT algorithm on input parameters p, $\tau(p)$ and $M(p)$ (cf. Definition 4). The function $f_s = f_{p,g,s}$ over $\mathbb{Z}_p$ is defined by

$$f_s(x) \stackrel{def}{=} P'_{p,s}(DL_{p,g}(x))$$

for all $x \in \mathbb{Z}_p^*$ and $f_s(0) = 0$. Note that we can access $f_s(x)$ for all $x \in S$ by querying $P'_{p,s}$ on $a = DL_{p,g}(x)$ provided in the advice. Our algorithm for $\mathrm{HNP}^{\mathcal{P},\varepsilon}$ follows.

```
Algorithm 1 Solving HNP^{P,ε}.

1. Run the SFT Algorithm 2 on input $p$,$\tau(p)$,$M(p)$, and oracle access
   to the restriction of $f_s$ to $S$; denote its output by $L$.
2. Output $((\alpha(p))^{-1} \cdot \beta)^{-1}$ for a uniformly random $\beta \in L$.
```

We show that Algorithm 1 outputs the hidden number s with probability $q(p) \geq \Omega(\tau(p))$. Fix p and denote $\alpha = \alpha(p)$, $\tau = \tau(p)$. Recall that $\left|\widehat{P_p}(\alpha)\right|^2 \geq \tau$ (since $\mathcal{P}$ is (M, τ, α)-concentrated), and that $\widehat{P_{p,s}}(\beta) = \widehat{P_p}(\beta s^{-1})$ $\forall \beta$ (by Proposition 1 Item 4 and the definition of $P_{p,s}(x) = P_p(s \cdot x)$). Therefore, the αs^{-1}-Fourier coefficient of $P_{p,s}$ is τ-significant, i.e.,

$$\left|\widehat{P_{p,s}}(\alpha \cdot s^{-1})\right|^2 \geq \tau.$$

Thus $L \ni \alpha s^{-1}$ with probability at least $1 - 1/p^{\Omega(1)}$ (by Theorem 4). Implying that

$$\beta = \alpha s^{-1}$$

with probability at least $(1 - 1/p^{\Omega(1)})/\left|L\right| \geq \Omega(\tau)$ (since β is a random element in L, and employing the bound $|L| \leq O(1/\tau)$ from Theorem 4). When $\beta = \alpha s^{-1}$, the output is

$$(\alpha^{-1}\beta)^{-1} = (\alpha^{-1}(\alpha s^{-1}))^{-1} = s.$$

We conclude that the output is s with probability $q(p) \geq \Omega(\tau)$.

Finally, the advice length $\ell(p)$ and the running time $t(p)$ are dominated by the query complexity and running time of the SFT Algorithm which is polynomial in $\log p$, $1/\tau(p)$ and $M(p)$ (cf. Theorem 4). □

Remark 1. Tighter bounds on the success probability $q(p)$ are possible at times. E.g., for the most significant bits $\mathcal{P} = \mathcal{MSB}_k$ for any $k \geq 1$, $q(p) \geq 1/2$.

3.2 Solving with Advice HNP$^{\mathcal{P},\varepsilon}$: Segment Predicates $\mathcal{P}$

We solve with advice HNP$^{\mathcal{P},\varepsilon}$ for segment predicates $\mathcal{P}$.

Let $\mathcal{P} = \left\{P_p \colon \mathbb{Z}_p^* \to \{\pm 1\}\right\}_{p \in \mathbb{P}}$. Let σ, a be functions mapping primes p to positive integers $\sigma(p)$ and to elements $a(p) \in \mathbb{Z}_p^*$. Denote by $\sigma(\mathcal{P})$ an upper bound on $\sigma(p)$ for all p.

Definition 3 (Segment Predicates [2]). *$\mathcal{P}$ is a (σ, a)-*segment predicate *if $\forall p$, $\exists P'_p \colon \mathbb{Z}_p^* \to \{\pm 1\}$ s.t.*

- *$P_p(x) = P'_p(x \cdot a(p))$ for all x, and*
- *$P'_p(x+1) \neq P'_p(x)$ for at most $\sigma(p)$ x's in $\mathbb{Z}_p$.*

$\mathcal{P}$ is a segment predicate *if $\exists c > 0$ s.t. $\sigma(p) < (\log p)^c$ for all p.*

We say that $\mathcal{P}$ is *far from constant* if $\exists \delta > 0$ s.t. $\forall p$, $\mathsf{maj}(P_p) \leq 1 - \delta$ for $\mathsf{maj}(P_p)$ the frequency of P_p's most common value.

Theorem 2. *Let $\mathcal{P}$ be a far from constant segment predicate and $\varepsilon \leq c/\sigma(\mathcal{P})$ for $c < 1$. Then there exists an algorithm that solves with advice HNP$^{\mathcal{P},\varepsilon}$.*

Proof. By Lemma 1, if $\mathcal{P}$ is a segment predicate, then $\mathcal{P}$ is concentrated; and furthermore, $\tau(\mathcal{P}) \geq 1/\sigma(\mathcal{P})$. By Theorem 1 this implies that there exists an algorithm that solves with advice HNP$^{\mathcal{P},\varepsilon}$. □

Lemma 1. *If $\mathcal{P}$ is a (σ, a)-segment predicate, then $\mathcal{P}$ is (M, τ, α)-concentrated for $M(p) = O(\sigma(p) \ln p)$, $\tau(p) = \Omega(1/\sigma(p))$, and $\alpha(p) = a(p)$.*

Proof. For each $P_p \in \mathcal{P}$, extend P_p to a function over $\mathbb{Z}_p$ by setting $P_p(0) = P_p(1)$. Fix p and drop its indices.

Consider first the case $a(p) = 1$. To show that $L_1(\widehat{P}) \leq M(p)$ and $\widehat{P}(1) \geq \tau(p)$, we first show that $\widehat{P}(\alpha) = \sum_{j=1}^{\sigma+1} (\ell_j/p) S_{\ell_j}(\alpha)$ for all $\alpha \in \mathbb{Z}_p$. A segment predicate with $a = 1$ defines a partition of $\mathbb{Z}_p$ into $\sigma + 1$ segments I_j, so that P is a constant $b_j \in \{\pm 1\}$ on each segment I_j. Thus, we can express P as a sum, $P = \sum_{j=1}^{\sigma+1} P_j$, of functions $P_j\colon \mathbb{Z}_p \to \{-1, 0, 1\}$ such that $P_j(x)$ is the constant $P(x)$ for $x \in I_j$ and 0 otherwise. By the linearity of the Fourier transform, for all $\alpha \in \mathbb{Z}_p$, $\widehat{P}(\alpha) = \sum_{j=1}^{\sigma+1} \widehat{P_j}(\alpha)$. By definition of the Fourier transform, $\widehat{P_j}(\alpha) = \frac{1}{p}\sum_{x\in I_j} b_j \chi_\alpha(x)$. Thus for c_j the starting point of I_j and $\ell_j = |I_j|$ its length, $\left|\widehat{P_j}(\alpha)\right| = |\chi_\alpha(c_j)| \left|\frac{1}{p}\sum_{x=0}^{\ell_j} \chi_\alpha(x)\right| = (\ell_j/p)S_{\ell_j}(\alpha)$ for $S_{\ell_j}(\alpha) = \frac{1}{\ell_j}\sum_{x=0}^{\ell_j - 1}\chi_\alpha(x)$ as defined in Proposition 2. We conclude that $\left|\widehat{P}(\alpha)\right| = \sum_{j=1}^{\sigma+1}(\ell_j/p)S_{\ell_j}(\alpha)$.

We show that $L_1(\widehat{P}) \le O(\sigma \ln p)$. By Proposition 2, $\left|S_{\ell_j}(\alpha)\right| \le O\left(\frac{p/\ell_j}{\mathsf{abs}(\alpha)}\right)$ for all ℓ_j, implying that $\left|\widehat{P}(\alpha)\right| \le \sum_{j=1}^{\sigma+1} O\left(\frac{(\ell_j/p)(p/\ell_j)}{\mathsf{abs}(\alpha)}\right) = O\left(\sigma/\mathsf{abs}(\alpha)\right)$. Thus, $L_1(\widehat{P}) = \sum_\alpha \left|\widehat{P}(\alpha)\right| \le O\left(\sigma \cdot \sum_\alpha \frac{1}{\mathsf{abs}(\alpha)}\right) = O(\sigma \ln p)$.

We show that $\left|\widehat{P}(1)\right| \ge \Omega(1/\sigma)$. Let ℓ_{j^*} be the length of the second longest segment in $I_1, \ldots, I_{\sigma+1}$. Clearly $\ell_{j^*} \le p/2$. Moreover, $\ell_{j^*} \ge \Omega(p/\sigma)$ because for far from constant $\mathcal{P}$, the longest segment is of length at most $(1-c)p$ for $c > 0$, implying that the second longest is of length at least the average length cp/σ over the remaining σ segments. By Proposition 2, $\left|S_\ell(1)\right|^2 \ge \Omega(1)$ for all $\ell \le p/2$. Thus, $\left|\widehat{P}_{j^*}(\alpha)\right|^2 \ge (\ell_{j^*}/p) \cdot \Omega(1) = \Omega(1/\sigma)$. We conclude that for $\alpha(p) = 1$ there is a function $\tau(p) \ge \Omega(1/\sigma(p))$ such that $\left|\widehat{P}(\alpha(p))\right|^2 \ge \tau(p)$ for all $p \in \mathbb{P}$.

Consider next the case of $a(p) \ne 1$. By definition of segment predicates, there exists P' s.t. $P(x) = P'(xa)$ for all $x \in \mathbb{Z}_p^*$. Extend P' to $\mathbb{Z}_p$. By Proposition 1, for all $\alpha \in \mathbb{Z}_p$ $\widehat{P(\alpha)} = \widehat{P}'(\alpha \cdot a^{-1})$. Implying that $L_1(\widehat{P}) = L_1(\widehat{P'}) \le O(\sigma \ln p)$ (because $\left\{\alpha a^{-1}\right\}_{\alpha\in\mathbb{Z}_p} = \mathbb{Z}_p$ for any a co-prime to p), and $\widehat{P}(a) = \widehat{P'}(a \cdot a^{-1}) = \widehat{P'}(1) \ge \Omega(1/\sigma)$.

We conclude that any family $\mathcal{P}$ of (σ, a)-segment predicates is (M, τ, α)-concentrated for $M(p) \le O(\sigma(p)\ln p)$, $\tau(p) \ge \Omega(1/\sigma(p))$ and $\alpha(p) = a(p)$. □

3.3 Solving with Advice HNP$^{\mathcal{P},\varepsilon}$: The Single Most Significant Bit

We solve with advice HNP$^{\mathcal{P},\varepsilon}$ for $\mathcal{P} = \mathcal{MSB}$ the *single* most significant bit.

Let $\mathcal{MSB} = \left\{MSB_p\colon \mathbb{Z}_p^* \to \{\pm 1\}\right\}_{p\in\mathbb{P}}$ the family of predicates giving the *single* most significant bit $MSB_p(x)$ of x (in a ±1 binary representation).

Theorem 3. *For any $\varepsilon = O(1)$ sufficiently small, there exists an algorithm that solves with advice HNP$^{\mathcal{MSB},\varepsilon}$.*

Proof. For the most significant bit MSB_p, $MSB_p(x+1) \ne MSB_p(x)$ only for one $x \in \mathbb{Z}_p^*$. Namely, $\mathcal{MSB}$ is a family of (σ, a)-segment predicates with $\sigma(p) = 1$, $a(p) = 1$ for all p. By Theorem 2, this implies that for any $\varepsilon = O(1)$ sufficiently small, there exists an algorithm that solves with advice HNP$^{\mathcal{P},\varepsilon}$. □

4 Universally Finding Significant Fourier Coefficients

In this section we present our universal SFT algorithm.

In the following We present the combinatorial condition on good queries sets S; show such sets exists; and prove that our SFT algorithm succeeds even when given oracle access only to the restriction of the input function f to the entries in S.

We define good queries. Recall that $A \subseteq \mathbb{Z}_N$ is *γ-biased* if $|\mathbb{E}_{x\in A}[\chi(x)]| < \gamma$ for all non-trivial characters χ of $\mathbb{Z}_N$. For $B, I \subseteq \mathbb{Z}_N$, we say that B is *(γ, I)-biased* if $|\mathbb{E}_{x\in B}[\chi(x)] - \mathbb{E}_{x\in I}[\chi(x)]| \leq \gamma$ for all characters χ is $\mathbb{Z}_N$. Denote by $A - B$ the set of differences $\{a-b\}_{a\in A, b\in B}$.

Definition 4 (Good Queries). *Let $\mathcal{S} = \{S_{N,\tau,t}\}_{N,\tau,t}$ be a family of sets $S_{N,\tau,t} \subseteq \mathbb{Z}_N$. We say that $\mathcal{S}$ is* good *if for all N, τ, t and for $\gamma = O(\tau/(t^2 \log N))$ sufficiently small, $S_{N,\tau,t} = \bigcup_{\ell=1}^{\lfloor(\log N)\rfloor}(A - B_\ell)$ s.t.*

- *A is γ-biased in $\mathbb{Z}_N$, of size $|A| = \Theta(\frac{1}{\gamma^2}\log N)$.*
- *$\forall\ell$, B_ℓ is $(\gamma, [0..2^\ell])$-biased in $\mathbb{Z}_N$, of size polynomial in $\log N$ and $1/\gamma$,*

We remark that the meaning of "sufficiently small γ" depends on the considered noise parameter ε, specifically, on the ratio $\varepsilon : \tau$. To simplify parameters, we fix this ratio to be, say, $\varepsilon < 0.9\tau$.

We show that good queries $\mathcal{S}$ exist. Moreover, there is a randomized algorithm that constructs good sets $S_{N,\tau,t}$ with high probability.

Proposition 4 (Good Queries Exist). *There is a randomized algorithm that given N, τ and t, outputs $S = S_{N,\tau,t}$ such that S is good with probability at least $1 - 1/N^{\Omega(1)}$; and its running time is $O(|S|)$.*

Proof. The algorithm outputs $S = \cup_{\ell=1}^{\lfloor(\log N)\rfloor}(A - B_\ell)$ for independent uniformly random sets $A \subseteq \mathbb{Z}_N$ and $B_\ell \subseteq [0..2^\ell]$, of sizes $|A| = O(\frac{1}{\gamma^2}\log N)$ and $|B_\ell| = O(\frac{1}{\gamma^2}\cdot\log N\cdot\log\log N)$, $\ell = 1, \ldots, \lfloor(\log N)\rfloor$.

Using Chernoff and Union bounds it is straightforward to show that S is good with probability at least $1 - 1/N^{\Omega(1)}$; details omitted. □

We show that our SFT algorithm succeeds when given oracle access to the restriction of the input function f (or its corruption by noise $f' = f + \eta$) to good queries $S = S_{N,\tau,t}$. Denote this restriction by $f'_{|S} \overset{def}{=} \{(x, f'(x))\}_{x\in S}$.

Let $\mathcal{S} = \{S_{N,\tau,t}\}$ be any family of good queries. For any integer $N > 0$, reals $\tau, t > 0$, a function $f\colon \mathbb{Z}_N \to \mathbb{B}_1$ s.t. $L_1(\widehat{f}) \leq t$, and an ε-random noise η for $\varepsilon < 0.9\tau$ the following holds.

Theorem 4 (SFT). *Our SFT algorithm, when given N, τ, t and $f'_{|S_{N,\tau,t}}$ for $f' = f + \eta$, outputs a list $L \supseteq \mathsf{Heavy}_\tau(f)$ of size $|L| \leq O(1/\tau)$, with probability at least $1 - 1/N^{\Omega(1)}$; and its running time is polynomial in $\log N$, $1/\tau$ and t.*

The probability is taken over the random noise η. In particular, when there is no noise, the success probability is 1.

Remark 2. Our SFT algorithm also handles: (i) Small amount of *adversarial noise*, that is, noise corrupting ε-fraction of the values of $f_{|S_{N,\tau,t}}$ for sufficiently small $\varepsilon = O(\tau/\log N)$. (ii) Input functions f accepting arbitrary complex values (and their corruption by noise f').

To prove Theorem 4, we first present the details of our SFT algorithm (Sect. 4.1), and then present its analysis (Sect. 4.2).

4.1 The SFT Algorithm

We give the details of our SFT algorithm. At a high level, the SFT algorithm is a binary search algorithm that repeatedly:

1. **Partitions** the set of potentially significant Fourier coefficients into two halves.
2. **Tests** each half to decide if it (potentially) contains a significant Fourier coefficient. This is done by **estimating** whether the sum of squared Fourier coefficients in each half exceeds the significance threshold τ.
3. **Continues recursively** on any half found to (potentially) contain significant Fourier coefficients.

At each step of this search, the set of potentially significant Fourier coefficients is maintained as a collection $\mathcal{J}$ of intervals: At the first step of the search, all Fourier coefficients are potentially significant, so $\mathcal{J}$ contains the single interval $J = [1..N]$. At each following search step, every interval $J \in \mathcal{J}$ is partitioned into two sub-intervals J_1 and J_2 containing the lower and upper halves of J respectively, and the set $\mathcal{J}$ is updated to hold only the sub-intervals that pass the test, i.e., those that (potentially) contain a significant Fourier coefficient. After $\log N$ steps this search terminates with a collection $\mathcal{J}$ of length one intervals revealing the frequencies of the significant Fourier coefficients. For all frequencies α of the significant Fourier coefficients, we then compute as an $O(\tau)$-approximation for $\widehat{f}(\alpha)$ the value $val_\alpha = \frac{1}{|A|}\sum_{x \in A-y} f(x)\overline{\chi_\alpha(x)}$ for some arbitrary $y \in \cup_{\ell=1}^{\lfloor(\log N)\rfloor} B_\ell$.

The heart of the algorithm is the test deciding which intervals potentially contain a significant Fourier coefficient (aka, distinguishing procedure). The distinguishing procedure we present, given an interval J, answers YES if its Fourier weight $weight(J) = \sum_{\alpha \in J} \left|\widehat{f}(\alpha)\right|^2$ exceed the significance threshold τ, and answers NO if the Fourier weight of a slightly larger interval $J' \supseteq J$ is less than $\tau/2$. This is achieved by estimating the ℓ_2 norm (i.e., sum of squared Fourier coefficients) of a filtered version of the input function f, when using a filter h that passes Fourier coefficients in J and decays fast outside of J.

The filters h that we use for depth ℓ of the search are the (normalized) *periodic square function* of support size 2^ℓ or Fourier domain translations of this function:

$$h_{\ell,c}(y) \stackrel{def}{=} \begin{cases} \frac{N}{2^\ell} \cdot \chi_{-c}(y) \; y \in [0..2^\ell] \\ 0 \qquad\qquad\quad otherwise \end{cases} \tag{1}$$

The filter $h = h_{\ell,c}$ passes all frequencies that lie within the length $N/2^\ell$ interval J centered around c, and decays fast outside of J. The filtered version of f is $f * h$, and we estimate its ℓ_2 norm $\|f * h\|_2^2$ by the estimator:

$$\mathsf{est}_{\ell,c}(f) \overset{def}{=} \frac{1}{|A|} \sum_{x \in A} \left(\frac{1}{|B_\ell|} \sum_{y \in B_\ell} \chi_{-c}(y) \overline{f(x-y)} \right)^2 \tag{2}$$

for $A, B_1, \dots, B_\ell \subseteq \mathbb{Z}_N$ as specified in the definition of good queries 4.

A pseudo-code of the algorithm follows. We denote intervals by the pair $\{a, b\}$ of their endpoints. To simplify notations, we assume: $(a' + b')/2$ is an integer (otherwise, appropriate flooring/ceiling is taken); $\|f\|_2 = 1$ (otherwise we normalize f it by dividing each read value by an energy estimator $\frac{1}{|A|} \sum_{x \in A} f(x)^2)$; $0 \in \bigcup_\ell B_\ell$ (otherwise we change variable in $\sum_{x \in A} \chi_\alpha(x) f(x)$ to $z = x - y$ for a random $y \in \bigcup_\ell B_\ell$).

```
Algorithm 2 SFT.
Input: $N \in \mathbb{N}$, $\tau \in (0, 1]$, $\{(x, y, f(x-y))\}_{x \in A, y \in B_\ell}$ $\forall \ell = 1, \dots, \lfloor (\log N) \rfloor$

1. Initialize: $\mathcal{J} \leftarrow \{\{0, N\}\}$
2. While $\exists \{a, b\} \in \mathcal{J}$ s.t. $b - a > 0$ do:
   (a) Delete $\{a, b\}$ from $\mathcal{J}$
   (b) For each pair $\{a', b'\}$ in $Low = \left\{a, \frac{a+b}{2}\right\}$, $High = \left\{\frac{a+b}{2} + 1, b\right\}$ do:
       i. Compute $\mathsf{est}_{\ell,c} \leftarrow \frac{1}{|A|} \sum_{x \in A} \left( \frac{1}{|B_\ell|} \sum_{y \in B_\ell} \chi_{-c}(y) f(x-y) \right)^2$ for $\ell = \log(N/(b' - a'))$, $c = \lfloor ((a' + b')/2) \rfloor$
      ii. If $\mathsf{est}_{\ell,c} \geq \tau/2$, insert $\{a', b'\}$ to $\mathcal{J}$
3. Sieving: For each $\{\alpha, \alpha\} \in \mathcal{J}$,
   (a) Compute $val(\alpha) \leftarrow \left| \frac{1}{|A|} \sum_{x \in A} \chi_\alpha(x) f(x) \right|^2$
   (b) If $val(\alpha) < \tau/2$, delete $\{\alpha, \alpha\}$ from $\mathcal{J}$
4. Output $L = \{\alpha \mid \{\alpha, \alpha\} \in \mathcal{J}\}$
```

4.2 Proof of Theorem 4

In this section we bring the proof of Theorem 4.

Proof of Theorem 4. Let $h_{\ell,c}$ and $\mathsf{est}_{\ell,c}(f)$ be as defined in (1)-(2). Fix a sufficiently small absolute constant $c > 0$. Consider condition (*) on $f' = f + \eta$:

$$(*) \quad \left| \mathsf{est}_{\ell,c}(f') - \|f * h_{\ell,c}\|_2^2 \right| < c\tau \text{ for all } \ell = 1, \dots, \lfloor (\log N) \rfloor, c \in \mathbb{Z}_N$$

By Lemma 2, when (*) holds, the SFT algorithm outputs $L \supseteq \mathsf{Heavy}_\tau(f)$ in running time polynomial in $\log N$, $1/\tau$ and t. By Lemma 3, when S is a good, (*) holds with probability at least $1 - 1/N^{\Omega(1)}$ over the noise η. Thus, the SFT algorithm outputs $L \supseteq \mathsf{Heavy}_\tau(f)$ in time polynomial in $\log N$, $1/\tau$ and t.

Proving $|L| \leq O(1/\tau)$ is similar. Consider condition $(*')$ saying that $\forall \alpha \in \mathbb{Z}_N$, $\left| \frac{1}{|A|} \sum_{x \in A} f'(x) \overline{\chi_\alpha(x)} - \widehat{f}(\alpha) \right| < c\tau$. We show that first, if (*') holds, then the sieving step leaves in $\mathcal{J}$ only $\{\alpha, \alpha\}$ s.t. $\left|\widehat{f}(\alpha)\right|^2 \geq \Omega(\tau)$; implying $|L| \leq O(1/\tau)$ by Parseval Identity. Second, when S is good, (*') holds with high probability over the noise η. We conclude that $|L| \leq O(1/\tau)$ with high probability over the noise η. Details omitted from this extended abstract. $\square$

We show that the SFT algorithm succeed on functions f' satisfying (*).

Lemma 2. *Let $f' = f + \eta$ and all other parameters be as in Theorem 4. If conditions (*) holds for f', then the SFT algorithm returns a list $L \supseteq \mathsf{Heavy}_\tau(f)$ in running time polynomial in $\log N$, $1/\tau$ and t.*

Proof. Denote $J = [a', b']$, $\ell = \log(N/(b' - a'))$ and $c = (a' + b')/2$.

Correctness. Consider a significant Fourier coefficient $\alpha \in \mathbb{Z}_N$. To show that $\alpha \in L$, it suffices to show that $\mathsf{est}_{\ell,c}(f') > \tau/2$ whenever $J \ni \alpha$. The latter is true because when J contains a τ-significant Fourier coefficient, then by Proposition 21 Item (1), $\|f * h_{\ell,c}\|_2^2 \geq \Omega(\sum_{\alpha \in J} \left|\widehat{f}(\alpha)\right|^2) \geq \Omega(\tau)$, which by (*) implies that $\mathsf{est}_{\ell,c}(f') \geq \Omega(\tau) \geq \tau/2$ (the latter holds by setting appropriate constants).

Efficiency. Fix ℓ, to bound the running time it suffices to show that "$\mathsf{est}_{\ell,c}(f') \geq \tau/2$" does not happen for too many disjoint intervals J of length $N/2^\ell$. If $\mathsf{est}_{\ell,c}(f') \geq \tau/2$, then by condition (*), $\|h_{\ell,c} * f\|_2^2 \geq \Omega(\tau)$. By Claim 21 Item 2, the latter implies that for a slightly larger interval $J' \supseteq J$, $|J'| / |J| \leq O(1/\gamma)$, its Fourier weight (that is, sum of squared Fourier coefficients with frequencies in J') is greater than $\Omega(\tau)$. This implies that $\mathsf{est}_{\ell,c}$ cannot be greater than $\tau/2$ too often, because there are at most $O(1/\tau)$ disjoint intervals whose Fourier weight exceeds $\Omega(\tau)$ (by Parseval Identity), and thus at most $O(\frac{1}{\tau} \cdot \frac{|J'|}{|J|})$ (possibly, overlapping) intervals J' whose Fourier weight exceeds $\Omega(\tau)$. $\square$

Claim 21 *For integers $\ell, c > 0$ and real $\gamma > 0$, let $J_{\ell,c} = \left\{\alpha \mid \mathsf{abs}(\alpha - c) \leq \frac{N}{2^\ell}\right\}$ an interval, and $J'_{\ell,c,\gamma} = \left\{\alpha \mid \mathsf{abs}(\alpha - c) \leq \sqrt{\frac{2}{3\gamma}} \cdot \frac{N}{2^\ell}\right\}$ its extension. Then: (1) $\|h_{\ell,c} * f\|_2^2 \geq \frac{1}{6} \sum_{\alpha \in J_{\ell,c}} \left|\widehat{f}(\alpha)\right|^2$, and (2) $\|h_{\ell,c} * f\|_2^2 \leq \sum_{\alpha \in J'_{\ell,c,\gamma}} \left|\widehat{f}(\alpha)\right|^2 + \gamma$.*

Proof. Denote $h = h_{\ell,c}$. By Parseval Identity and the convolution theorem, $\|h * f\|_2^2 = \sum_\alpha \left|\widehat{h}(\alpha)\right|^2 \left|\widehat{f}(\alpha)\right|^2$. By definition of h, $\widehat{h}(\alpha) = S_{2^\ell}(\alpha - c)$ for $S_t(\alpha) = \frac{1}{t} \sum_{y=0}^{t-1} \chi_\alpha(y)$ as defined in Proposition 2. The proof follows from the properties guaranteed in Proposition 2; details omitted from this extended abstract. $\square$

We show that when using a good set of queries S condition (*) holds (with high probability over the random noise η).

Lemma 3. *Let $f' = f + \eta$ and all other parameters be as in Theorem 4. Condition (*) holds for f' with probability at least $1 - 1/N^{\Omega(1)}$ over the noise η.*

Proof. Let $S = \bigcup_{\ell=1}^{\log N} (A - B_\ell)$ for $S = S_{N,t,\tau}$ from the good queries $\mathcal{S}$ of Theorem 4. Recall that A is a γ-biased set and the B_ℓ's are $(\gamma, [0..2^\ell])$-biased.

Fix $\ell \in [\lfloor (\log N) \rfloor]$ and $c \in \mathbb{Z}_N$. Denote $B = B_\ell$, $h = h_{\ell,c}$. Observe that $\left|\mathsf{est}_{\ell,c}(f') - \|h * f\|_2^2\right| \leq (i) + (ii) + (iii)$ for:

- $(i) := \left|\mathsf{est}_{\ell,c}(f) - \|h * f\|_2^2\right|$
- $(ii) := \left|2\frac{1}{|A|}\sum_{x\in A}\left(\frac{1}{|B|}\sum_{y\in B}\chi_{-c}(y)f(x-y)\right)\left(\frac{1}{|B|}\sum_{y\in B}\chi_{-c}(y)\eta(x-y)\right)\right|$
- $(iii) := |\mathsf{est}_{\ell,c}(\eta)|$

We bound each of these terms. By Claim 22, $(i) \le O(\gamma L_1(\widehat{f})^2 \log N)$. By Claims 23-24, with probability at least $1 - 3\exp(-\Omega(|A|\,\tau^2))$, $(ii) + (iii) \le (2 + O(\gamma L_1(\widehat{f})^2 \log N))(2\varepsilon^2 + \varepsilon + O(\tau))$. Thus, for $\gamma = O(\tau/(t^2 \log N))$ and $\varepsilon = O(\tau)$, with probability at least $1 - 3\exp(-\Omega(|A|\,\tau^2))$,

$$\left|\mathsf{est}_{\ell,c}(f') - \|h * f\|_2^2\right| \le O(\tau) \text{ for all } f \text{ s.t. } L_1(\widehat{f}) \le t.$$

By union bound, this holds for *all* $\ell = 1, \ldots, \lfloor(\log N)\rfloor$ with probability at least $1 - 3\exp(-\Omega(|A|\,\tau^2))\log N = 1 - 1/N^{\Omega(1)}$ since $|A| \ge \Omega((\ln N)/\tau^2)$ by definition of good sets. □

Claim 22 *$(i) \le O(\gamma L_1(\widehat{f})^2 \log N)$.*

Proof. Denote $I = [0..2^\ell]$. Define $g_x(y) = \chi_{-c}(y)\overline{f(x-y)}$ for $y \in I$ and $g_x(y) = 0$ otherwise. Then by the definition of $\mathsf{est}_{\ell,c}(f)$ and $\|h * f\|_2^2$,

$$(i) = \left|\mathop{\mathbb{E}}_{x\in A}\left(\mathop{\mathbb{E}}_{y\in B} g_x(y)\right)^2 - \mathop{\mathbb{E}}_{x\in\mathbb{Z}_N}\left(\mathop{\mathbb{E}}_{y\in I} g_x(y)\right)^2\right| \le (i') + (ii') \quad \text{for:}$$

- $(i') := \left|\mathbb{E}_{x\in A}\left(\mathbb{E}_{y\in B}\, g_x(y)\right)^2 - \mathbb{E}_{x\in A}\left(\mathbb{E}_{y\in I}\, g_x(y)\right)^2\right|$
- $(ii') := \left|\mathbb{E}_{x\in A}\left(\mathbb{E}_{y\in I}\, g_x(y)\right)^2 - \mathbb{E}_{x\in \mathbb{Z}_N}\left(\mathbb{E}_{y\in I}\, g_x(y)\right)^2\right|$

We show below that $(i') \le \gamma \cdot L_1(\widehat{f})^2 \cdot O(\log N)$ and $(ii') \le \gamma \cdot L_1(\widehat{f})^2$. Combining these bounds we get that $(i) \le O(\gamma L_1(\widehat{f})^2 \log N)$.

Bounding term (i'). We first get rid of the expectation over $x \in A$ by upper bounding it with its value on a maximizing $x_0 \in A$. We then switch to the Fourier representation of g_{x_0} and rely on B being (γ, I)-biased to bound the difference between the expectations over $y \in B$ and $y \in I$. Finally, we bound the emerging quantity $L_1(\widehat{g_{x_0}})$ (using Proposition 2 and algebraic manipulations). Details omitted from this extended abstract.

Bounding term (ii'). We first observe that the inner expectations are over the same range I and variable. That is, $(ii') = |\mathbb{E}_{x\in A}\, \bar{g}(x) - \mathbb{E}_{x\in\mathbb{Z}_N}\, \bar{g}(x)|$ for $\bar{g}(x) = \left(\mathbb{E}_{y\in I}\, g_x(y)\right)^2$. We then switch to the Fourier representation of $\bar{g}$ and rely on A being γ-biased to bound the difference between the expectations over $x \in A$ and $x \in \mathbb{Z}_N$.

$$(ii') \le \sum_{\alpha\in\mathbb{Z}_N}\left|\widehat{\bar{g}}(\alpha)\right|\left|\mathop{\mathbb{E}}_{x\in A}\chi_\alpha(x) - \mathop{\mathbb{E}}_{x\in\mathbb{Z}_N}\chi_\alpha(x)\right| \le \gamma L_1(\widehat{\bar{g}})$$

Finally we bound the emerging quantity $L_1(\widehat{g})$. Observe that $\bar{g} = (h * f)^2$ (since $h * f = \mathbb{E}_{y \in \mathbb{Z}_N} \frac{N}{|I|} \chi_{-c}(y)\overline{f(x-y)} = \mathbb{E}_{y \in I}\, \chi_{-c}(y)\overline{f(x-y)})$. Therefore, $L_1(\widehat{g}) \leq L_1(\widehat{h * f})^2$ where we use the fact that for any function s, $L_1(\widehat{s^2}) \leq L_1(\widehat{s})^2$. Observe further that $L_1(\widehat{h * f})^2 \leq L_1(\widehat{f})^2$ because $\left|\widehat{h * f}(\alpha)\right| = \left|\widehat{h}(\alpha)\right| \cdot \left|\widehat{f}(\alpha)\right| \leq \left|\widehat{f}(\alpha)\right|$, where the last inequality follows since $\left|\widehat{h}(\alpha)\right| \leq 1$ for all α. Combining the above bounds we conclude that $(ii') \leq \gamma L_1(\widehat{f})^2$. □

Claim 23 $(ii) \leq (1 + O(\gamma L_1(\widehat{f})^2 \log N))(2\varepsilon^2 + \varepsilon + O(\tau))$ *with probability at least* $1 - \exp\left(-\Omega(|A|\,\tau^2)\right)$.

Proof. By Cauchy-Schwartz inequality, $(ii)^2 \leq 4 \cdot (a) \cdot (b)$ for

- $(a) := \frac{1}{|A|} \sum_{x \in A} \left(\frac{1}{|B_\ell|} \sum_{y \in B_\ell} \chi_{-c}(y) f(x-y) \right)^2$
- $(b) := \frac{1}{|A|} \sum_{x \in A} \left(\frac{1}{|B_\ell|} \sum_{y \in B_\ell} \chi_{-c} \eta(x-y) \right)^2$.

To bound (b), observe that $(b) = \mathsf{est}_{\ell,c}(\eta) \leq (iii)$. Therefore, by Claim 24, $(b) \leq 2\varepsilon^2 + \varepsilon + O(\tau)$ with probability at least $1 - 2\exp(-\Omega(|A|\,\tau^2))$.

To bound (a), observe that $(a) = \mathsf{est}_{\ell,c}(f)$, implying by Claim 22 that $\left|(a) - \|h * f\|_2^2\right| \leq O(\gamma L_1(\widehat{f})^2 \log N)$. Next observe that $\|h * f\|_2^2 \leq 1$ (since $\|h * f\|_2^2 = \sum_\alpha \left|\widehat{h}(\alpha)\widehat{f}(\alpha)\right|^2$ where $\left|\widehat{h}(\alpha)\right|, \left|\widehat{f}(\alpha)\right| \leq 1$ for all α).[5] We conclude therefore that $|(a)| \leq 1 + O(\gamma L_1(\widehat{f})^2 \log N)$.

Combining both bounds we conclude that with probability at least $1 - \exp\left(-\Omega(|A|\,\tau^2)\right)$, $(ii) \leq (1 + O(\gamma L_1(\widehat{f})^2 \log N))(2\varepsilon^2 + \varepsilon + O(\tau))$. □

Claim 24 $(iii) \leq 2\varepsilon^2 + \varepsilon + O(\tau)$ *with probability at least* $1 - 2\exp(-\Omega(|A|\,\tau^2))$.

Proof. To bound $(iii) = |\mathsf{est}_{\ell,c}(\eta)|$ we rely on the randomness of η. By definition of $\mathsf{est}_{\ell,c}(\eta)$ and the triangle inequality, $|\mathsf{est}_{\ell,c}(\eta)| \leq \frac{1}{|A|} \sum_{x \in A} \left(\frac{1}{|B|} \sum_{y \in B} |\eta(x-y)| \right)^2$. Opening the parenthesis, $|\mathsf{est}_{\ell,c}(\eta)| \leq (a) + (b)$ for:

- $(a) := \frac{1}{|A|} \sum_{x \in A} \frac{1}{|B|^2} \sum_{y_1 \neq y_2 \in B} |\eta(x-y_1)|\,|\eta(x-y_2)|$
- $(b) := \frac{1}{|A|} \sum_{x \in A} \frac{1}{|B|^2} \sum_{y \in B} |\eta(x-y)|^2$

Expressions (a) and (b) are averages over the indep. random variables: $v_{x,y_1,y_2} = |\eta(x-y_1)|\,|\eta(x-y_2)| \cdot \frac{|B|}{|B|-1}$ and $v_{x,y} = |\eta(x-y)|^2 \cdot \frac{1}{|B|}$ respectively (the factors involving $|B|$ are for proper normalization). We use Chernoff/Hoeffding bound to upper bound expressions (a) and (b) separately, and then apply union bound to upper bound their sum. Details omitted from this extended abstract. □

[5] Here, $\left|\widehat{f}(\alpha)\right| \leq 1$ because f accepts values in $\mathbb{B}_1$. The bound holds also for unbounded f, provided f is normalized to have $\sum_\alpha \left|\widehat{f}(\alpha)\right|^2 \leq 1$.

5 Bit Security Implications

We obtain bit security results as a corollary of our algorithm solving $\mathrm{HNP}^{\mathcal{P},\varepsilon}$.

We set some terminology. Let $G = \{g_p\}$ be a family of generators g_p of $\mathbb{Z}_p^*$. Let $\mathcal{F} = \{f_p\}$ be a family of functions f_p outputting secrets s when given public data $PD_{p,g,s}$ depending on the modulus p, a generator g and the secret s. Think of $\mathcal{F}$ as the underlying hard to compute function. Let $\mathcal{P} = \{P_p\}$ be a family of predicates over $\mathbb{Z}_p^*$. Denote by MB a "magic box" that, given p, g and $PD_{p,g,s}$, outputs $MB(p, g, PD_{p,g,s}) \overset{def}{=} P_p(s)$. We say that:

- *$\mathcal{P}$ is as hard as $\mathcal{F}$* if there is an algorithm A that, given $PD_{p,g_p,s}$, oracle access to MB, and an advice depending only on p and g_p, outputs the secret s with probability at least $1/poly(\log p)$, while the running time and advice length are polynomial in $\log p$.
- $\mathcal{F}$is *$\mathcal{G}$-accessible* if there is an access algorithm that, given public data $PD_{p,g,s}$ for a secret s, and an element $a \in \mathbb{Z}_{p-1}$, outputs public data $PD_{p,g,s\cdot g^a}$ for the secret $s \cdot g^a \bmod p$.

Theorem 5. *For any $\mathcal{G}$-accessible $\mathcal{F}$ and concentrated $\mathcal{P}$, $\mathcal{P}$ is as hard as $\mathcal{F}$.*

Proof. Fix p and denote $g = g_p$. Let $Adv_{p,g}$ be an advice depending only on p and g as used in Theorem 1 for solving $\mathrm{HNP}^{\mathcal{P},\varepsilon}$ in Algorithm 1. Let $P_{p,s}(a) \overset{def}{=} P_p(s \cdot g^a)$. Observe that given $PD_{p,g,s}$ and oracle access to MB we can simulate oracle access to $P_{p,s}$: For each query a, we compute $PD_{p,g,s\cdot g^a}$ using the access algorithm of $\mathcal{F}$, and output $val = MB(p, PD_{p,g,s\cdot g^a})$. By definition of MB, $val = P_p(s \cdot g^a)$.

The algorithm A runs Algorithm 1 while simulating oracle access to $P_{p,s}$. By Theorem 1, the output is s with probability at least $1/poly(\log p)$. We conclude that $\mathcal{P}$ is as hard as $\mathcal{F}$. □

Let $\mathcal{OK}$ and $\mathcal{EL}'$ denote the underlying hard families of functions in the Okamoto conference key sharing scheme and in the (modified) ElGamal public key encryption scheme as defined in [5]. The analysis of [5] shows that $\mathcal{OK}$ ($\mathcal{EL}'$) is $\mathcal{G}$-accessible. We conclude therefore that for any concentrated predicate $\mathcal{P}$, $\mathcal{P}$ is as hard as computing $\mathcal{OK}$ ($\mathcal{EL}'$). In particular, this holds for $\mathcal{P} = MSB_1$.

Acknowledgments.

The author is grateful to Shafi Goldwasser, Igor Shparlinski, Vinod Vaikuntanathan, and Avi Wigderson for helpful discussions.

References

1. A. Akavia. *Learning Noisy Characters, Multiplication Codes and Cryptographic Hardcore Predicates.* PhD dissertation; defended Aug 2007, MIT, EECS, Feb 2008.

2. A. Akavia, S. Goldwasser, and S. Safra. Proving Hard-Core Predicates using List Decoding. In *Proc. of 44th IEEE Annual Symposium on Foundations of Computer Science (FOCS'03)*, pages 146–157. IEEE Computer Society, 2003.
3. Adi Akavia. Finding significant fourier coefficients deterministically and locally. ECCC Report TR08-102, 2008.
4. D. Boneh and R. Venkatesan. Hardness of computing the most significant bits of secret keys in diffie-hellman and related schemes. *Lecture Notes in Computer Science*, 1109:129–142, 1996.
5. D. Boneh and R. Venkatesan. Rounding in lattices and its cryptographic applications. In *SODA: ACM-SIAM Symposium on Discrete Algorithms (A Conference on Theoretical and Experimental Analysis of Discrete Algorithms)*, 1997.
6. A. C. Gilbert, S. Guha, P. Indyk, S. Muthukrishnan, and M. Strauss. Near-optimal sparse fourier representations via sampling. In *Proc. of 34 ACM Annual Symposium on Theory of Computing (STOC'02)*, pages 152–161. ACM Press, 2002.
7. A. C. Gilbert, S. Muthukrishnan, and M. Strauss. Improved time bounds for near-optimal sparse fourier representation via sampling. In *in Proc. SPIE Wavelets XI*, 2005.
8. Maria Isabel González-Vasco and Igor Shparlinski. On the security of diffie-hellman bits. In *Proc. Workshop on Cryptography and Computational Number Theory, Singapore 1999, Birkhäuser*, pages 257–268, 2001.
9. W. Hoeffding. Probability inequalities for sums of bounded random variables. *J. Amer. Stat. Assoc*, 58:13–30, 1963.
10. M. A. Iwen. A deterministic sub-linear time sparse fourier algorithm via non-adaptive compressed sensing methods. *CoRR*, abs/0708.1211, 2007.
11. M. A. Iwen. A deterministic sub-linear time sparse fourier algorithm via non-adaptive compressed sensing methods. In *SODA*, pages 20–29, 2008.
12. E. Kushilevitz and Y. Mansour. Learning decision trees using the Fourier spectrum. *SICOMP*, 22(6):1331–1348, 1993.
13. A. K. Lenstra, H. W. Lenstra, and L. Lovsz. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):515–534, 1982.
14. Igor Shparlinski and Arne Winterhof. A nonuniform algorithm for the hidden number problem in subgroups. In Feng Bao, Robert H. Deng, and Jianying Zhou, editors, *Public Key Cryptography*, volume 2947 of *Lecture Notes in Computer Science*, pages 416–424. Springer, 2004.
15. Igor Shparlinski and Arne Winterhof. A hidden number problem in small subgroups. *Math. Comp.*, 74:2073–2080, 2005.
16. Audrey Terras. *Fourier Analysis on Finite Groups and Applications*. Cambridge U. Press, Cambridge, U.K., 1999.